THE UNIVERSITY OF QUEENSLAND

AUSTRALIA

Action Analysis and Video Summarisation to Efficiently Manage and Interpret Video Data

Johanna Carvajal

B.Eng., M.Eng.

A thesis submitted for the degree of Doctor of Philosophy at The University of Queensland in 2016

School of Information Technology and Electrical Engineering

Abstract

In the last few years we have seen how the volume of video data has exponentially grown. Specialised online sites like YouTube and NetFlix are attracting a considerable amount of audience who are uploading, accessing, and actively interacting with the online sites. Furthermore, millions of video surveillance cameras have been installed around the world. Video cameras are installed to monitor shopping centres, universities, parks, streets, and in general to monitor any public place. Undoubtedly, it is becoming indispensable to efficiently and automatically manage and interpret all the massive amount of video data available nowadays. Computer vision is the science responsible for processing images and videos. The main goal of this thesis is to contribute towards efficiently managing and interpreting video data via action analysis and video summarisation. Action analysis using computer vision techniques is essential given that the majority of the available videos contain human actions. Action analysis is a broad topic that covers several areas. For instance, we can find: action recognition, joint action segmentation and recognition, and action assessment.

For the action recognition problem, there are several techniques designed to recognise actions. Among them, two schools of thoughts have gained attention recently. On one hand, traditional video encoders and its variants are the main reference for action recognition. Traditional video encoders include the popular Bag of Visual Words and the Fisher Vector representation. On the other hand, statistical modelling of actions via Riemannian manifolds offers an interesting alternative to traditional video encoders. To this end, we provide a detailed analysis of the performance of the two aforementioned schools of thoughts for action recognition under same set of features across several datasets. The detailed analysis also investigates when these methods break and how performance degrades when the datasets have challenging conditions, likely to be encountered in uncontrolled situations.

To address the joint action segmentation and recognition problem, we propose two hierarchical systems where a given video is processed as a sequence of overlapping temporal windows. Both proposed systems require fewer parameters to be optimised and avoid the need for a custom dynamic programming definition as in previous works. The last action analysis problem this thesis focuses on is action assessment. Action assessment is still in early stages. Action assessment consists in assessing how well people perform actions. Learning how to automatically assess actions can be a valuable tool. For instance, catwalk competitions require human assessment which may be highly subjective. However, to date, nobody has attempted to apply computer vision techniques to automatically assess the quality of how someone strides down the catwalk.

Action analysis is not the only way to process video information. Video summarisation is an active area of research within the computer vision community. Instead of tedious manual review of hours and hours of video, video summarisation aims to provide a concise and informative summary of the video. We present a novel approach to video summarisation that makes use of a Bag-of-visual-Textures approach which is computationally efficient and effective. Our approach can be used for short-term and long-term videos. On long-term videos the proposed system considerably reduces the amount of footage with only minor degradation in the information content.

Declaration by author

This thesis is composed of my original work, and contains no material previously published or written by another person except where due reference has been made in the text. I have clearly stated the contribution by others to jointly-authored works that I have included in my thesis.

I have clearly stated the contribution of others to my thesis as a whole, including statistical assistance, survey design, data analysis, significant technical procedures, professional editorial advice, and any other original research work used or reported in my thesis. The content of my thesis is the result of work I have carried out since the commencement of my research higher degree candidature and does not include a substantial part of work that has been submitted to qualify for the award of any other degree or diploma in any university or other tertiary institution. I have clearly stated which parts of my thesis, if any, have been submitted to qualify for another award.

I acknowledge that an electronic copy of my thesis must be lodged with the University Library and, subject to the policy and procedures of The University of Queensland, the thesis be made available for research and study in accordance with the Copyright Act 1968 unless a period of embargo has been approved by the Dean of the Graduate School.

I acknowledge that copyright of all material contained in my thesis resides with the copyright holder(s) of that material. Where appropriate I have obtained copyright permission from the copyright holder to reproduce material in this thesis.

Publications during candidature

1. Johanna Carvajal, Chris McCool, and Conrad Sanderson. **Summarisation of Short-term and Long-Term Videos using Texture and Colour**. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 769-775, 2014 [25].

2. Johanna Carvajal, Chris McCool, Conrad Sanderson, and Brian C. Lovell. **Multi-Action Recognition via Stochastic Modelling of Optical Flow and Gradients**. In *PRICAI Workshop on Machine Learning for Sensory Data Analysis (MLSDA)*, pp. 19–24, 2014 [26]. Awarded ***Best Paper Award***.

3. Johanna Carvajal, Chris McCool, Brian C. Lovell, and Conrad Sanderson. **Joint Recognition and Segmentation of Actions via Probabilistic Integration of Spatio-Temporal Fisher Vectors**. In *Trends and Applications in Knowledge Discovery and Data Mining*, ser. Lecture Notes in Computer Science, vol. 9794, pp. 115-127, 2016 [24].

4. Johanna Carvajal, Arnold Wiliem, Chris McCool, Brian C. Lovell, and Conrad Sanderson. **Comparative Evaluation of Action Recognition Methods via Riemannian Manifolds, Fisher Vectors and GMMs: Ideal and Challenging Conditions**. In *Trends and Applications in Knowledge Discovery and Data Mining*, ser. Lecture Notes in Computer Science, vol. 9794, pp. 88–100, 2016 [27].

5. Johanna Carvajal, Arnold Wiliem, Conrad Sanderson, and Brian C. Lovell. **Towards Miss Universe Automatic Prediction: The Evening Gown Competition**. Accepted for publication at the *International Conference on Pattern Recognition (ICPR)*, 2016 [28].

Publications included in this thesis

1. Johanna Carvajal, Arnold Wiliem, Chris McCool, Brian C. Lovell, and Conrad Sanderson. **Comparative Evaluation of Action Recognition Methods via Riemannian Manifolds, Fisher Vectors and GMMs: Ideal and Challenging Conditions**. In *Trends and Applications in Knowledge Discovery and Data Mining*, ser. Lecture Notes in Computer Science, vol. 9794, pp. 88–100, 2016 [27]. Incorporated within Chapter 5.

Contributor	Statement of contribution
Johanna Carvajal (Candidate)	Conception and design of algorithm (70%) Design of experiments (80%) Paper writing (80%)
Arnold Wiliem	Conception and design of algorithm (15%) Design of experiments (10%) Paper review
Chris McCool	Paper review
Brian C. Lovell	Paper review
Conrad Sanderson	Conception and design of algorithm (15%) Design of experiments (10%) Paper writing and editing (20%)

2. Johanna Carvajal, Chris McCool, Conrad Sanderson, and Brian C. Lovell. **Multi-Action Recognition via Stochastic Modelling of Optical Flow and Gradients**. In *PRICAI Workshop on Machine Learning for Sensory Data Analysis (MLSDA)*, pp. 19–24, 2014 [26]. Awarded ***Best Paper Award***. Incorporated within Chapter 6.

Contributor	Statement of contribution
Johanna Carvajal (Candidate)	Conception and design of algorithm (70%) Design of experiments (80%) Paper writing (80%)
Chris McCool	Conception and design of algorithm (10%) Design of experiments (10%) Paper writing and editing (10%)
Conrad Sanderson	Conception and design of algorithm (20%) Design of experiments (10%) Paper writing and editing (10%)
Brian C. Lovell	Paper review

3. Johanna Carvajal, Chris McCool, Brian C. Lovell, and Conrad Sanderson. **Joint Recognition and Segmentation of Actions via Probabilistic Integration of Spatio-Temporal Fisher Vectors**. *Trends and Applications in Knowledge Discovery and Data Mining*, ser. Lecture Notes in Computer Science, vol. 9794, pp. 115-127, 2016 [24]. Incorporated within Chapter 6.

Contributor	Statement of contribution
Johanna Carvajal (Candidate)	Conception and design of algorithm (70%) Design of experiments (80%) Paper writing (70%)
Chris McCool	Conception and design of algorithm (15%) Paper review
Brian C. Lovell	Paper review
Conrad Sanderson	Conception and design of algorithm (15%) Design of experiments (20%) Paper writing and editing (30%)

4. Johanna Carvajal, Arnold Wiliem, Conrad Sanderson, and Brian C. Lovell. **Towards Miss Universe Automatic Prediction: The Evening Gown Competition**. Accepted for publication at the *International Conference on Pattern Recognition (ICPR)*, 2016 [28]. Incorporated within Chapter 7.

Contributor	Statement of contribution
Johanna Carvajal (Candidate)	Conception and design of algorithm (80%) Design of experiments (90%) Dataset creation (100%) Paper writing (70%)
Arnold Wiliem	Conception and design of algorithm (20%) Design of experiments (10%) Paper writing and editing (20%)
Conrad Sanderson	Paper writing and editing (10%)
Brian C. Lovell	Paper review

5. Johanna Carvajal, Chris McCool, and Conrad Sanderson. **Summarisation of Short-term and Long-Term Videos using Texture and Colour**. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 769-775, 2014 [25]. Incorporated within Chapter 9.

Contributor	Statement of contribution
Johanna Carvajal (Candidate)	Conception and design of algorithm (60%) Design of experiments (70%) Paper writing (80%)
Chris McCool	Conception and design of algorithm (10%) Design of experiments (10%) Paper writing and editing (10%)
Conrad Sanderson	Conception and design of algorithm (30%) Design of experiments (20%) Paper writing and editing (10%)

Contributions by others to the thesis

The work contained in this thesis was carried out by the author under the guidance and supervision of her advisors, Professor Brian C. Lovell and Dr. Conrad Sanderson. Part of the work contained in this thesis was carried out by the author under the collaboration and discussions with Dr. Chris McCool and Dr. Arnold Wiliem.

Statement of parts of the thesis submitted to qualify for the award of another degree

None.

Acknowledgements

This thesis is dedicated to all those new or potential PhD students. As many of you, one day I had the dream to obtain a PhD degree from a top university. At the beginning, I seemed to be extremely difficult, but thanks to my determination and perseverance it was possible to accomplish my dream. This was not an easy journey and it took me many years to get here. However, when I am getting to the end on this journey I can assure all you that all my efforts were worthy. This dream would not be possible without the support of many people that I would like to acknowledge.

I would like to express my sincere gratitude to my supervisors Dr. Conrad Sanderson and Prof. Brian Lovell for trusting and supporting me during this journey. I was very fortunate to have also the collaboration and advice of Dr. Chris McCool and Dr. Arnold Wiliem. All of them provided me different points of view that helped me through every situation that I faced during my candidature. Besides my advisors, I am also grateful to my PhD colleagues in the Advance Surveillance team who were sharing with me this amazing experience and all what it implied: candidature milestones, conference deadlines, frustration, paper rejection, etc.

I also appreciate the financial support received from The University of Queensland and NICTA by awarding the UQ Centennial Scholarship, the UQ International Fees, and NICTA top-up scholarship.

I am deeply thankful with my lovely family. Specially thanks to my parents and my brother for reminding me every day of what I am capable of. To my beloved brother Mauricio who passed away at such early age, you are always in our thoughts.

Finally, all my gratitude to my partner Jhon who has been always by my side encouraging, supporting and loving me. Thanks for joining me in this Australian adventure, more than 16,000 kilometres away from home. I would not be able to successfully cross the PhD line without your daily care and motivation.

Thank you God for all your blessings

Keywords

action analysis, human action recognition, catwalk analysis, action segmentation, mixture of Gaussians, Riemannian manifolds, Fisher vectors, video summarisation, keyframe selection, texture information

Australian and New Zealand Standard Research Classifications (ANZSRC)

ANZSRC code: 080104, Computer Vision, 60%
ANZSRC code: 080109, Pattern Recognition and Data Mining 40%

Fields of Research (FoR) Classification

FoR code: 0801, Artificial Intelligence and Image Processing, 100%

Contents

List of Figures

List of Tables

Acronyms and Abbreviations

2D/3D	2/3 Dimensional
BoT	Bag of Textures
BoVW	Bag of Visual Words
ConvNet	Convolutional Neural Network
CNN	Convolutional Neural Networks
DCG	Discounted Cumulative Gain
DCT	Discrete Cosine Transform
DT	Dense Trajectories
FV	Fisher Vector
GMM	Gaussian Mixture Model
HoG	Histogram of Gradients
HoF	Histogram of Optical Flow
HMM	Hidden Markov Model
HMM-MIO	HMM with Multiple Irregular Observations
DCG	Ideal Discounted Cumulative Gain
IDT	Improved Dense Trajectories
KDE	Kernel Density Estimation
LibLinear	Library for Large Linear Classification
LibSVM	Library for Support Vector Machines

LS	Linear Subspaces
LSA	Localised Soft Assignment
MBH	Motion Boundary Histogram
MULR	Miss Universe Listwise Ranking
MUPR	Miss Universe Pairwise Ranking
NDCG	Normalized Discount Cumulative Gain
NN	Nearest Neighbour
OpenCV	Open source Computer Vision
Poly	Polynomial
PI-FV	Probabilistic Integration with FV
PI-GMM	Probabilistic Integration with GMM
P-CNN	Pose-based CNN
PCA	Principal Component Analysis
RBF	Radial Basis Function
RKHS	Reproducing Kernel Hilbert Space
RP	Random Projection
SA	Soft Assignment
SFV	Stacked Fisher Vector
SIFT	Scale Invariant Feature Transform
SR	Sparse Representation
SURF	Speeded Up Robust Features
SPC	Sparse Coding
SPD	Symmetric Positive Matrices
SVM	Support Vector Machines
VLAD	Vector of Locally Aggregated Descriptors

Mathematical Notation

$\boldsymbol{v}$	a vector (lower-case)
$\boldsymbol{v}^\top$	transpose of vector $\boldsymbol{v}$
$[v_1, v_2, \cdots, v_D] = [v_i]_{i=1}^D$	contents of a D-dimensional row vector
$\|\|\boldsymbol{v}\|\| = \sqrt{v_1^2 + v_2^2 + \cdots + v_D^2}$	norm of vector $\boldsymbol{v}$
$\boldsymbol{A}$	a matrix (upper-case, boldface)
$\boldsymbol{A}^\top$	transpose of matrix $\boldsymbol{A}$
$\mathcal{N}(\boldsymbol{x}\|\boldsymbol{\mu}, \boldsymbol{\Sigma})$	a Gaussian distribution with $\boldsymbol{\mu}$ as the mean and $\boldsymbol{\Sigma}$ as the covariance matrix
$\boldsymbol{A}^{-1}$	inverse of matrix $\boldsymbol{A}$
$\|\boldsymbol{A}\|$	determinant of matrix $\boldsymbol{A}$
$\|\|\boldsymbol{A}\|\|_F^2$	Frobenius norm on $\boldsymbol{A}$
$\log(\boldsymbol{A})$	Matrix logarithm
$\{\boldsymbol{X}_i\}_{i=1}^n$	set of n elements: $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_n$
$\mathbf{1}_{\mathcal{W}_s}(t)$	indicator function, resulting in 1 if $t \in \mathcal{W}_s$, and 0 otherwise
$\text{sign}(\cdot)$	signum function extracts the sign of a number

Part I

Preliminaries

Chapter 1

Introduction

Life is not easy for any of us. But what of that? We must have perseverance and above all confidence in ourselves. We must believe that we are gifted for something, and that this thing, at whatever cost, must be attained.

Marie Curie

According to *Cisco Systems, Inc.*, online video data around the world will be responsible from 80% of all consumer internet traffic in 2019 [4]. This percentage represents a 64% increase with respect to 2014. Moreover, the sum of all forms of video will be in the range of 80% to 90% of consumer traffic by 2019 [4]. The internet video traffic include videos from sites such as YouTube (short-form videos), Hulu (long-term videos), Netflix (internet video to TV), BitTorrent (video exchanged through peer-to-peer file sharing), Vudu (online video purchases and rentals), and also videos from webcam views and web-based video monitoring [4]. Online video data is not the only existing video data. Every day millions of hours of video are captured around the world by surveillance cameras. There were an estimated 245 million globally installed surveillance cameras which were active and operational in 2014, according to IHS Technology [5].

With the huge amount of video data currently available and its predicted increase in the next few years, it is necessary to develop intelligent automatic systems able to efficiently analyse, process, and interpret the information contained in the video data. Video data can be efficiently managed in several manners using computer vision techniques. This thesis presents two ways to efficiently manage video information:

1. **Action analysis**: Given that a massive part of video data contains humans, analysis of human action has become a hot topic in recent years in the computer vision community.

2. **Video Summarisation**: Instead of tedious manual review of hours and hours of video, video summarisation aims to provide a concise and informative summary of the video.

1.1 Goals and Challenges

As mentioned before, the amount of online video data and video captured by surveillance cameras available nowadays is massive and is expected to keep growing. The main goal of this thesis is to contribute towards efficiently managing and interpreting video information via action analysis and video summarisation.

Action analysis covers several areas. For instance, we can find: action recognition, joint action segmentation and recognition, and action assessment. Many approaches have been developed to recognise single human actions, in the midst of them there are two schools of thoughts: *(i)* traditional video encoding techniques and *(ii)* statistical modelling of actions via Riemannian manifolds.

The most traditional approach for video encoding is the Bag-of-Visual Words (BoVW) [116, 140, 169]. In the BoVW approach feature descriptors are quantised into visual words using a visual vocabulary. The visual vocabulary is typically generated via k-means [174]. A video is then characterised as the frequency histogram over visual words [140]. Among other video encoders we can find: Soft Assignment (SA), Localised Soft Assignment (LSA), Sparse Coding (SPC), Vector of Locally Aggregated Descriptors (VLAD), and Fisher Vector (FV). Particularly, the FV approach has been successfully applied to action recognition in recent times [112, 117, 168].

Simultaneously, there has been a growing interest in solving the action recognition problem using Riemannian manifolds [67, 95, 161]. Two widely used statistics for modelling actions are: covariance matrices, which are naturally Symmetric Positive Definite (SPD), and linear subspaces (LS).

Both schools of thoughts, traditional video encoders and Riemannian manifolds, have shown competitive performance. However, it is still an open question which school of thought best describes and recognises human actions under the same set of features across several datasets. Between the Riemannian representation based on SPD matrices and LS, it is still unknown which modelling best represents human actions. Moreover, it is also currently unknown which school of thought is the most robust and capable of dealing with challenges present in realistic and uncontrolled scenarios.

There are other areas of action analysis that have not been widely investigated or are in early stages and need more attention for the sake of creating reliable automatic systems. For example, the joint action segmentation and recognition is one of these areas that has been less explored. The action segmentation and recognition problem, in the context of this thesis, consists of segmenting and recognising continuous actions from a video, where one person performs a sequence of several single actions [142]. This is an important problem given that in natural and realistic settings of human behaviour, the fundamental problem is segmenting and recognising actions from a sequence containing several single actions [21].

Another attractive area of research for action analysis is action assessment. The assessment of quality of actions using only visual information is still under early development. A recent work to predict the expert judges' scores for diving and figure skating in the Olympic games is presented in [121]. The concept behind the score prediction is to learn how to assess the quality of actions in videos. This concept can open the door to reveal other ways where the assessment of an action

can be a valuable tool. For instance, catwalk competitions have been fashionable for a long time. However, to date, nobody has attempted to apply computer vision techniques to assess the quality of how someone strides down the catwalk.

Apart from action analysis, video data information can be also managed using video summarisation techniques. Video summarisation is an active area of research within the computer vision community and it has been applied to provide summaries in various video categories such as wildlife videos [181], sports videos [113], TV documentaries [11], among others. Video summarisation, also known as still image abstraction, static storyboard or static video abstract, is a compilation of representative frames selected from the original video [39]. Video summarisation still faces the challenges of creating a useful, intuitive, and informative summary [102]. It often deals with the problem of key frame selection: which key frames should be preserved in the output summary? [50]

1.2 Contributions

This thesis presents a series of contributions to address the following two tasks: action analysis and video summarisation. Below a brief overview of the contributions is given. Throughout the chapters comprised in this thesis more details are given that further explain each of the following contributions.

1.2.1 Comparative Evaluation of Action Recognition Approaches

- We provide a detailed analysis of performance of the traditional video encoding techniques and the alternative Riemannian manifolds methods under the same set of features across several datasets.
- We employ two categories of Riemannian manifolds: symmetric positive matrices and linear subspaces. For both categories we use their corresponding nearest neighbour classifiers, kernels, and recent kernelised sparse representations.
- We quantitatively show when these methods break and how the performance degrades when the datasets have challenging conditions (translations and changes in scale).

1.2.2 Joint Action Recognition and Segmentation

- We propose two novel hierarchical systems to perform action segmentation and recognition where a given video is processed as a sequence of overlapping temporal windows.
- The proposed methods are based on GMMs and the FV representation.
- The combination of probabilistic integration either with FVs or GMM is novel for the action segmentation and recognition problem.
- Our proposed method based on FV outperforms one existing approach and it is much faster than the GMM approach.

- The proposed systems require fewer parameters to be optimised and avoid the need for a custom dynamic programming definition as in previous works.

1.2.3 Catwalk Analysis (Action Assessment)

- We are the first to assess the quality of how someone strides down the catwalk using computer vision techniques.
- We propose a novel dataset called the Miss Universe (MU) dataset that comprises 10 years of the Miss Universe evening gown competition.
- We study two novel problems for automatic ranking the catwalk of each participant during the Miss Universe evening gown competition.
- The first sub-problem is The Miss Universe Listwise Ranking (MULR) problem. It aims to predict the winner of the evening gown competition.
- The second sub-problem is The Miss Universe Pairwise Ranking (MUPR) problem. It focuses on judging the catwalk between two participants.
- We propose an approach that addresses both problems simultaneously.
- We adapt recent video descriptors, shown to be effective in action recognition, into our framework.

1.2.4 Video Summarisation

- We present a novel approach to summarise videos that makes use of a Bag-of-visual-Textures (BoT) approach which is computationally efficient and effective.
- We first propose the use of texture information to improve video summarisation.
- Two systems are proposed, one based solely on the BoT approach and another which exploits both colour information and BoT features.
- We show how our approach can be used for short-term and long-term videos.
- Proposed system reduces the amount of footage in long-term videos by a factor of 27, with only minor degradation in the information content.

1.3 Thesis Outline

This section provides an outline of the entire thesis. The rest of this thesis is comprised of 3 major parts: Action Analysis, Video Summarisation, and Final Remarks.

Part II: Action Analysis

- **Chapter 2: Literature Review**. This chapter overviews the literature review for action analysis. To start with, the chapter describes popular video descriptors for action analysis. The chapter then depicts how video descriptors can be either encoded or statistical modelled via Riemannian manifolds. Finally, various approaches for single action recognition, action segmentation and recognition, and catwalk assessment are delineated.

- **Chapter 3: Background Theory**. This chapter equips the reader with the relevant theory used for Part II. It first describes the video descriptors used in this work, followed by the definitions of GMMs, the FV representation, and Riemannian manifolds.

- **Chapter 4: Datasets for Action Recognition**. The existing datasets used for action analysis are reported in this chapter. The datasets included are: KTH, UCF-Sports, UT-Tower, and CMU-MMAC.

- **Chapter 5: Comparative Evaluation of Action Recognition Techniques**. This chapter presents a comparative evaluation of various techniques for action recognition while keeping as many variables as possible controlled. Two categories of Riemannian manifolds are employed: symmetric positive matrices and linear subspaces. This chapter also compares against traditional action recognition techniques based on GMMs and FVs. These action recognition techniques are evaluated under ideal conditions, as well as their sensitivity in more challenging conditions.

- **Chapter 6: Joint Recognition and Segmentation of Actions**. This chapter presents two hierarchical approaches that perform joint classification and segmentation. For the first approach, a given video is processed via a sequence of overlapping temporal windows. Each frame in a temporal window is represented through selective low-level spatio-temporal features. Features from each window are represented as a FV. Instead of directly classifying each FV, it is converted into a vector of class probabilities. The second proposed approach is based on GMMs. This GMM approach also processes a given video via a sequence of overlapping temporal windows. The vector of class probabilities for the GMM approach is obtained using the average log-likelihood over each temporal window. For both proposed approaches, the final classification decision for each frame is then obtained by integrating the class probabilities at the frame level, which exploits the overlapping of the temporal windows. Experiments were performed on two datasets: s-KTH (a stitched version of the KTH) and the challenging CMU-MMAC dataset.

- **Chapter 7: Towards Miss Universe Automatic Prediction via Catwalk Analysis**. This chapter analyses if we can predict the winner of Miss Universe after watching how they stride down the catwalk during the evening gown competition. As this problem has not been investigated before, we analyse whether existing computer vision approaches can be used to automatically extract the qualities exhibited by the Miss Universe winners during their catwalk. This study can pave the way towards new vision based applications for the fashion industry. We propose a novel video dataset, called the Miss Universe dataset, collected from the evening gown competition. To describe the videos we employ the recently proposed Stacked Fisher Vectors.

Part III: Video Summarisation

- **Chapter 8: Literature Review**. This chapter concisely defines video summarisation including its main characteristics. An overview of various common approaches for video summarisation is also provided.

- **Chapter 9: Summarisation of Short-Term and Long-Term Videos**. This chapter presents a novel approach to video summarisation that makes use of a Bag-of visual-Textures (BoT) approach. Two systems are presented, one based solely on the BoT approach and another which exploits both colour information and BoT features. On short-term videos we show that our BoT and fusion systems both achieve state-of-the-art performance. When applied to a new underwater surveillance dataset containing 33 long-term videos, the proposed system reduces the amount of footage with only minor degradation in the information content.

Part IV: Final Remarks

- **Chapter 10: Overall Main Findings**. This chapter summarises the main contributions of the research.

- **Chapter 11: Potential Future Work**. This chapter enumerates new avenues and improvements for extension of the work.

Part II

Action Analysis

Chapter 2

Literature Review

Disciplining yourself to do what you know is right and important, although difficult, is the highroad to pride, self-esteem, and personal satisfaction.

Margaret Thatcher

Action analysis has potential applications in smart homes, building surveillance systems, human action retrieval, biometric gait recognition, and action assessment. In smart houses action recognition may be used for assisted living such as healthcare, lifestyle analysis, security and surveillance, and interaction monitoring [45].

In surveillance systems it is crucial to monitor human behaviour to detect unusual or suspicious events [83, 92]. The detection can be done "after-effect" or in real-time [109]. Video action retrieval aims to retrieve similar video-content, but instead of searching using textual information (keywords), visual features are extracted to find a subset of videos with similar content [127, 152]. Biometric gait recognition in conjunction with surveillance cameras aims to identify subjects that may pose a risk. Gait recognition can be done unobtrusively as both the capture and the recognition are carried out at a considerable distance from the subjects [170]. Gait and action assessments have been investigated for fall risk assessment for older adults, for the rehabilitation of humans with neurological disorders, and assessment of functional mobility [49, 166, 153].

The assessment of quality of actions using only visual information is still under development. A recent work to predict the expert judges' scores for diving and figure skating in the Olympic games is presented in [121]. The concept behind the score prediction is to learn how to assess the quality of actions in videos. This concept can open the door to reveal other ways where the assessment of an action can be a valuable tool. For instance, catwalk competitions have been fashionable for a long time. However, to date, nobody has attempted to apply computer vision techniques to assess the quality of how someone strides down the catwalk.

The majority of existing frameworks for action analysis encompass three main steps [149]: video descriptors, video encoders (or dictionary learning) to form a representation for a video based on the video descriptors , and finally classification of the video using the representation, e.g. support

vector machines (SVM). This chapter summarises the state-of-the-art for the two main stages of action analysis: video descriptors and video encoders in Sections 2.1 and 2.2, respectively.

Video encoders are not the only method to model actions. Another school of thought uses statistical modelling instead of traditional video encoders to capture the variability of actions. Many statistical models have a natural geometric structure that lies in the Riemannian geometry. Popular approaches employed for action analysis using this geometry are compiled in Section 2.3.

Finally, Section 2.4 summarises the overall literature review for the three problems covered in this thesis which are related to action analysis: *(i)* single action recognition, *(ii)* joint action segmentation and recognition, and *(iii)* catwalk assessment. In consideration of the fact that catwalk assessment has not been investigated before, some closely-related areas can provide an idea of how this problem can be solved.

2.1 Video Descriptors

Several descriptors have been employed to represent human actions. For example, Histograms of Gradients (HoG) and Histograms of Optical Flow (HoF) were introduced to characterise local motion and appearance [85]. The histograms are calculated from spatial gradients and optical flow accumulated in space-time neighbourhoods of detected interest points. Scale Invariant Feature Transform (SIFT) [96] has been also employed. SIFT transforms an image into a large collection of local feature vectors, each of which is invariant to image translation, scaling, and rotation, and partially invariant to illumination changes and affine or 3D (3-dimensional) projection. SIFT features are largely invariant to changes in scale, illumination, and local affine distortions. Another existing descriptor is the Speeded Up Robust Features (SURF) [15] which is a scale- and rotation-invariant interest point detector and descriptor SURF is robust to motion blur, partly inspired by SIFT. Motion boundary histograms (MBH) descriptor was presented in [37] for human detection by computing derivatives separately for the horizontal and vertical components of the optical flow. MBH is more robust to camera motion than optical flow, and thus more discriminative for action recognition [167].

Fusing descriptors has become also popular for action recognition. For instance, fusion of motion and shape prototypes has been investigated in [70]. An action is represented as a sequence of prototypes which can be an efficient and flexible approach for action matching in long video sequences.

Combining the benefits of gradients and optical flow is another way of fusing descriptors [139, 54]. Gradients have been used as a relatively simple yet effective video representation [129]. Each pixel in the gradient image helps extract relevant information, eg. edges of a subject. Gradients can be computed at every spatio-temporal location (x, y, t) in any direction in a video. Since the task of action recognition is based on an ordered sequence of frames, optical flow can be used to provide an efficient way of capturing local dynamics and motion patterns in a scene [53].

Another approach that fuses descriptors with the purpose of capturing the local motion information of the video trajectories is presented in [167, 168]. The following descriptors are then extracted and aligned with the trajectories: point coordinates (shape), histograms of oriented gradients (ap-

pearance), histograms of optical flow (motion), and motion boundary histograms (differential optical flow). This descriptor is best known as dense trajectories (DT).

Improved dense trajectories (IDT) were later proposed to remove trajectories caused by camera motion. To this end, feature points between frames using SURF descriptors and dense optical flow are matched. These matches are used to estimate a homography and then rectify the image to remove the camera motion.

In recent times, convolutional neural networks (CNNs or ConvNets) [88] have gained attention for the action analysis [34, 155, 76, 148]. Inspired by the success in the image-domain, CNNs have been extended to large-scale video classification in [76]. This proposed extension processes input video at two streams to improve the runtime performance of CNNs: a low-resolution context stream and a high-resolution fovea stream that only operates on the middle portion of the frame.

Two-Stream Convolutional Networks technique [148] is a temporal ConvNet on optical flow trained in order to capture the information on appearance not only from still frames but also from motion between frames. The results in [148] are significantly better than training on raw stacked frames as in [76]. Spatio-temporal features are learnt using deep 3D ConvNet [155]. 3D ConvNets enclose the information provided by objects, scenes, and actions in videos, showing superior performance thn the 2D ConvNet features, similar to the work presented in [76].

Other descriptors are built fusing successful existing descriptors. For example, the trajectory-pooled deep-convolutional descriptor fuses the benefits of DT [167, 168] and two-stream networks [148]. Using convolutional feature maps and improved trajectories, the final descriptor is obtained pooling the local ConvNet responses over the spatio-temporal tubes centred at the trajectories.

The goal of all the aforementioned action representations is to recognise full-body activities such as walking or jumping. However, not long ago researches have realised the importance of recognising small differences in activities such as cut and peel in food preparation or dry hair and brushing hair in common daily activities [34, 77, 122, 134]. Pose-based CNN descriptor (P-CNN) tracks human body parts for recognising the small differences in actions [34]. The descriptor aggregates motion and appearance information along the tracks [34]. Moreover, P-CNN descriptors are also combined with DT by late fusion of the individual classification scores achieving an improvement in the recognition performance [34].

Recurrent Neural Network (RNN) can also model the temporal evolution of features in video [74]. RNN combined with CNN is explored to model the spatio-temporal evolution of human facial expressions in [74]. However, the majority of the techniques involving CNNs or RNNs deal with video as flat data sequence, overlooking the hierarchical structure of the video content [90]. To alleviate this situation, a video is represented by a hierarchical structure in [90]. The video is fragmented from small to large. The small fragments contain single frames, followed by consecutive frames to capture motion, then short video clips, and finally the large fragment contains the entire video. Long Short-Term Memory (LSTM) networks are applied on the different fragments to exploit long-term temporal dynamics [14, 106, 108].

LSTM is also employed in conjunction with CNN [106], where several convolutional temporal feature pooling architectures are explored. CNNs are then connected to LSTM cells to model the video as an ordered sequence of frames. RNN architecture with one hidden layer of LSTM cells is proposed in [14]. To this end, 3D CNNs are used to automatically learn spatio-temporal features. A RNN is then trained to classify human actions without using any prior knowledge. Experimental results showed that the introduction of the LSTM classification improves system performance. Object detection is used jointly with LSTM for fine grained action detection [108]. To this end, an interactional object parsing method based on LSTM is used.

Despite the recent popularity of RNNs and techniques derived from it (e.g. LSTM), the obtained networks are usually seen as black boxes [151]. It is difficult to uncover their mechanism of operation. Moreover, due to the difficulty to decipher these networks, the ability to design better configurations is limited [75]. In other words, it is uncertain the source of their notable performance and their shortcomings [75]. Furthermore, neural networks rely on large annotated datasets, with hundreds of samples for each category [81, 114]. For action analysis, current datasets are still too small and noisy [46]. Consequently, results obtained should be analysed with prudence as concluded in [46]. For the aforementioned reasons, there is an open dilemma that makes researchers wonder whether similar RNN mechanisms can be also determined and employed by these networks in real-world data [75].

2.2 Video Encoders

Under the influence of feature encoding methods employed for object recognition [30, 116], video encoders are also being efficiently employed for action analysis [116, 140, 169].

The conventional approach to encode features is the Bag-of-Visual Words (BoVW). In the BoVW approach, feature descriptors are quantised into visual words using a visual vocabulary. The visual vocabulary is generated via k-means [174]. A video is then characterised as the frequency histogram over the visual word [140]. Latter day advances substitute the hard quantisation of feature descriptors associated with the BoVW with other encodings that hold more information about the original descriptors [30]. Substitution methods for hard assignment encodings include: Soft Assignment (SA), Localised Soft Assignment (LSA), Sparse Coding (SPC), Fisher Vector (FV), and Vector of Locally Aggregated Descriptors (VLAD).

SA was first introduced in [160] for scene categorisation. Instead of frequency histogram, SA employs a kernel density estimation (KDE) for estimating a probability density function. KDE uses a kernel function with a smoothing factor that controls the softness of the assignment [160, 174]. For each local feature, the k-th coefficient represents the degree of membership of that local feature being to the k-th visual word [160]. LSA which was developed in [94] is similar to SA, but only considering the k nearest visual words into encoding. SPC was originally proposed for image classification [180]. In SPC, selective sparse coding are used instead of traditional vector quantization to extract salient properties of local visual descriptors.

High dimensional encoding frameworks such as FV and VLAD are the newest trend for feature encoding. Both, VLAD and FV are exhibiting significant improvements on challenging datasets for action recognition [112, 177, 116, 168]. Alike BoVW, the visual vocabulary in VLAD is learnt using k-means. Each local descriptor is then associated to its nearest visual word, and the differences between the local descriptors and the visual words are accumulated [68].

FV encodes additional information [36, 168]. Rather than encoding the frequency of the descriptors, as for BoVW, FV encodes the deviations from a probabilistic version of the visual dictionary. This is done by computing the gradient of the sample log-likelihood with respect the parameters of the dictionary model. The dictionary model is usually generated via GMMs. The parameters are the zero, first and second order statistics. Since more information is extracted, a smaller visual dictionary size can be used than for BoF, in order to achieve the same or better performance.

2.3 Statistical Modelling of Video Action Descriptors via Riemannian Manifolds

The dynamic information within the feature descriptors can be statistically modelled by exploring correlations and variations among feature descriptos [95]. Two widely used statistics are: covariance matrices, which are naturally Symmetric Positive Definite (SPD), and linear subspaces (LS). The SPD matrices and LS of the Euclidean space are known to lie on Riemannian manifolds, where the underlying distance metric is not the usual l_2 norm [161, 67].

SPD matrices have been used to describe gesture and action recognition in [44, 54, 139]. Grassmann manifolds, which are special cases of Riemannian manifolds, represent a set of m-dimensional linear subspaces and have also been investigated for the action recognition problem [97, 98, 99, 111]. The straightforward way to deal with Riemannian manifolds is via the nearest-neighbour (NN) scheme. For SPD matrices, NN classification using the log-Euclidean metric for covariance matrices is employed in [159, 54]. Canonical or principal angles are used as a metric to measure similarity between two LS and have been employed in conjunction with NN in [159].

Manifolds can be also mapped to a reproducing kernel Hilbert space (RKHS) by using kernels. Kernel analysis on SPD matrices and LS has been used for gesture and action recognition in [60, 66, 146, 161]. SPD matrices are embedded into RKHS via a pseudo kernel in [60]. With this pseudo kernel is possible to formulate a locality preserving projections over SPD matrices. Positive definite radial kernels are used to solve the action recognition problem in [66], where an optimisation algorithm is employed to select the best kernel among the class of positive definite radial kernels on the manifold.

An improved Grassmann discriminant analysis based on Grassmann kernels and a graph-embedding framework is presented [146]. Recently, the traditional sparse representation (SR) on vectors has been generalised to sparse representations in SPD matrices and LS [55, 59, 57, 165]. While the objective of SR is to find a representation that efficiently approximates elements of a signal class with as few

atoms as possible, for the Riemannian SR, any given point can be represented as a sparse combination of dictionary elements [59, 57].

In [57], LS are embedded into the space via isometric mapping, which leads to a closed-form solution for updating a LS representation, atom by atom. Moreover, [57] presents a kernelised version of the dictionary learning algorithm to deal with non-linearity in data. The sparse coding and dictionary learning problem for SPD matrices are outlined in [59]. To this end, SPD matrices are embedded into the RKHS to perform sparse coding.

2.4 Action Analysis Approaches

2.4.1 Single Action Recognition

Hidden Markov models (HMMs) have been used in conjunction with shape-context features to recognise single actions [101]. The shape-context features are extracted using the image contours and dividing the region where the person is into uniform tiles. For each tile a feature vector is generated. The discrete cosine transform is used in order to minimise redundancy and to compress the data. Continuous HMM with mixed Gaussian output probability is employed with a simple left to right topology.

Gaussian Mixture Models (GMMs) have also been explored for the single-action detection and classification. For the approach presented in [92], each action is represented by a combination of GMMs. Each action is modelled by two sets of feature attributes. The first set represents the change of body size, while the second represents the speed of the action. Features with high correlations for describing actions are grouped into the same Category Feature Vector (CFV). All CFVs related to the same category are then modelled using a GMM. A Confident-Frame-based Recognising algorithm is used for recognition, where the video frames which have high confidence are used as a specialised model for classifying the rest of the video frames.

A video sequence can be also represented as a BoVW [169]. This approach and its variants are among the most popular approaches for action recognition [168, 169]. The standard approach consists of four main steps: low-level feature extraction, offline codebook generation, feature encoding and pooling, and normalisation. The offline codebook, also known as visual dictionary, is constructed using k-means clustering of a large training dataset. Visual words are then defined as the centers of the clusters, with the size of visual dictionary equal to the number of the clusters. To encode low-level features, each feature vector is assigned to its closest dictionary word using Euclidean distance. A given video is then represented as a frequency histogram over the visual words. Normalised histograms can then be used by linear classifiers.

FV has been used for a wide range of applications in the computer vision community, such as image retrieval [42], as well as image representation and classification [36, 135]. Recently, FV has been also successfully applied to the single-action recognition problem [112, 168]. Descriptors based on MBH and SIFT are used in conjunction with FV in [112]. The dimensionality of each descriptor is

reduced to 64 dimensions via Principal Component Analysis (PCA). A similar approach is presented in [168], where the SIFT descriptors are replaced by histograms of gradients, optical flow, and trajectories. To combine various descriptor types, FVs derived from each descriptor type are concatenated.

The action recognition problem can be also solved by decomposing actions into sub-actions or atoms [171, 172]. In the former, motion atoms are obtained using a discriminative clustering method. These atoms are basic units used to construct motion phrase with a longer scale. To this end, a bottom-up phrase construction algorithm and a greedy selection method are used. Motion phrase is composed of multiple atomic motion units. The latter work presents a latent hierarchical model. This hierarchical model has a tree structure, where each node represents a sub-action. Two latent variables are then used to represent the starting and ending time points of each sub-action.

2.4.2 Action Segmentation and Recognition

In contrast to single action recognition, relatively less work exists on action segmentation and recognition. The action segmentation and recognition problem, in our context, consists of segmenting and recognising continuous actions from an image sequence (video), where one person performs a sequence of several single actions [142]. See Fig. 6.1 for an example of a video containing a sequence of several single actions. The process for segmenting and recognising multiple actions in a video can be solved either as two independent problems or as a joint problem.

As an independent problem, two segmentation techniques based on colour intensity and motion are employed to partition a video containing continuous actions in indoor scenes [141]. Both techniques efficiently segment periodic human movements and count the number of action cycles. The segmentation is evaluated in a new dataset created to replicate a sport center or a gym. To count the number of action cycles, two existing indoor datasets are used, which do not contain continuous actions, simply several cycles of the same action. Having the actions segmented, actions can be recognised using motion and shape features.

One of the first methods to jointly segment and recognise actions is called Multi-Task Conditional Random Field (MT-CRF) and was proposed in [143]. This method consists of classifying motions into multi-labels, e.g. a person folding their arms while sitting. This approach has been only applied on two synthetic datasets. Two methods [19, 63] have been applied to realistic multi-action datasets. Hoai et al. [63] deal with the dual problem of human action segmentation and classification. The recognition model is trained using labelled data with a multi-class SVM. Once the model for all actions has been obtained, the video segmentation and recognition is done using dynamic programming, maximising the SVM score of the winning class while suppressing those of the non-maximum classes. The feature mapping depends on the dataset employed, and includes trajectories, features extracted from binary masks, and STIPs.

The method proposed in [63] has several drawbacks. One drawback is the requirement of fully labelled annotations for training. Furthermore, it suffers from the limitations of dynamic programming where writing the code that evaluates sub-problems in the most efficient order is often nontrivial [164]. Also, the binary masks are not always available and the STIP descriptors have deficiencies. STIP

based descriptors are computationally expensive, unstable, imprecise and can result in unnecessarily sparse detections [22, 53]. This method also requires an extensive search for optimal parameters.

An approach termed Hidden Markov Model for Multiple, Irregular Observations (HMM-MIO) [19] has also been proposed for the action recognition and segmentation task. HMM-MIO jointly segments and classifies observations which are irregular in time and space, and are characterised by high dimensionality. The high dimensionality is reduced by probabilistic PCA. Moreover, HMM-MIO deals with heavy tails and outliers exhibited by empirical distributions by modelling the observation densities with the long-tailed Student's t distribution.

HMM-MIO requires the search of the following four optimal parameters: **(i)** the resulting reduced dimension, **(ii)** the number of components in each observation mixture, **(iii)** the degree of the t-distribution, and **(iv)** the number of cells (or regions) used to deal with space irregularity. As feature descriptors, HMM-MIO extracts STIPs, with the default 162-dimensional descriptor. HMM-MIO hence suffers from the drawback of a large search of optimal parameters and the use of STIP descriptors.

An algorithm for temporally segmenting videos into atomic movements using a Bayesian framework is presented in [163]. This Bayesian framework is tested on a dataset of interactive movements. A dataset of the interactive movements of *reach* and *strike* is used for this purpose. For the interactive movement *reach*: the subjects were asked to pick up and place the objects on different surfaces. For movement *reach* the subjects perform actions such as: stepped around, bent, used either of their hands. For the interactive movement *strike*: the subjects were asked to strike and throw objects placed at different heights. For movement *strike* the subjects perform actions such as: punched, slammed down and slapped (forehand and backhand).

Differences to Related Work on Action Recognition and Segmentation: There are similar works to the action recognition and segmentation problem that computer vision techniques have also attempted to solve such as action detection, event detection, and skeletal action segmentation and recognition. Each of them will be briefly explained including how those differ from the action recognition and segmentation problem as defined in this thesis.

Action detection is different from action recognition and segmentation. Action detection aims to find if an action is performed in a large video [48]. In particular, the work presented in [48] uses the *Coffee and Cigarettes* dataset collected from a single movie and is composed of 11 short stories with various scenes and actors. The goal is to correctly localise or detect the 41 drinking and 70 smoking examples within the dataset.

Action-oriented event detection is also a distinctive problem from the action segmentation and recognition problem. An event is formed combining various subjects, actions, scenes, and objects (e.g. John and Laura are climbing a mountain on a sunny day) [84]. While the goal of action-event detection is to identify the temporal range of an event in a video and may also include the location [33], the goal of action segmentation and recognition is to segment and recognise short or long videos of an actor executing an unidentified action (e.g. climb) into one of several classes [79].

Motion capture (MoCap) is the process of capturing only the people's movements, not his or her visual appearance [47, 100]. Various works also deal with the action recognition and segmentation problem using MoCap [51] or similar hardware to acquire the human skeleton such as RGB-D cameras (Microsoft Kinect) [82, 176]. However, given that the skeleton is used instead of the the visual information, those approaches are not directly comparable with the action recognition and segmentation problem as defined in this work.

2.4.3 Catwalk Assessment

Catwalk assessment using computer vision techniques has not been investigated yet. However, there are some related areas that can give some clues on how to solve this problem. For instance, gait analysis, action assessment for sports, and fine-grained action analysis are close-related areas.

Gait analysis and action assessment have been investigated for various applications. The approaches for gait and action assessment are being evaluated in datasets captured using specialised equipment such as Kinect and multi-view cameras. Kinect and multi-view cameras have been employed for fall risk assessment, humans with neurological disorders, asymmetric gait, and assessment of functional mobility [166, 49, 153, 107]. Two web-cams are used to extract gait parameters including walking speed, step time, and step length in [166]. The gait parameters are used for a fall risk assessment tool for home monitoring of older adults.

For rehabilitation and treatment of patients with neurological disorders, automatic gait analysis with a Microsoft Kinect sensor is used to quantify the gait abnormality of patients with multiple sclerosis [49]. Different pose estimations are dynamically modelled using the continuous-state HMM to describe and assess the quality of four motions used by clinicians to assess functional mobility [153]. The assessment of functional mobility is important for patients with musculoskeletal disorders, where it is necessary to determine between abnormal and normal movements. The quality of movement is evaluated in a dataset acquired using Kinect skeleton data. The four motions under considerations are: walking on a flat surface, gait on stairs, and transitions between sitting and standing.

A gait analysis for asymmetric gait recognition is presented in [107]. The asymmetry between the left and right body parts is calculated in order to facilitate the gait assessment. The asymmetric gait system consists of two camcoders located on the right and left side of a treadmill. This system fully reconstructs the skeleton model and demonstrates good accuracy compared to Kinect sensors.

Although kinect and multi-view cameras provided valuable information, both sources of recording data are not always available. Lately, the assessment of quality of actions using only visual information is gaining attention.

Recently, a work for action assessment was presented in [121]. The action assessment approach trains a regression model using pose and discrete cosine transform (DCT) features. Due to the lack of dataset to evaluate the quality of an action, a new dataset was collected from online video depicting the Olympics games and other worldwide competitions, where the judge's scores were public. Two sport categories were collected: *diving* and *figure skating*.

A similar work to [121] uses the estimated pose for each frame to obtain the approximate entropy features [162]. The goal is to quantify the quality of diving actions. An SVM regressor is then trained to generate real-valued scores as an indicative of the quality of diving actions. The diving subset of [121] is also used. The experimental results show that entropy-based feature performs better than the traditional DCT-based feature employed in [121].

Action assessment has also found applicability to determine the expertise level of surgical skills of medical students [185]. To this end, a time-series is generated from motion features extracted from video data. Frequency coefficients are then computed and the nearest neighbour approach is used to classify among three levels surgical skills (beginner, intermediate, and expert). Given that there was not an available dataset for this problem, 18 medical students were recruited to acquire the dataset. For each student, two instances for the tasks of *suturing* and *knot tying* were collected and an expert evaluated the skills of each participant.

Catwalk analysis can be also related to fine-grained action analysis. Fine-grained action analysis has been recently investigated for action recognition [34, 77, 122, 134], where it is important to recognise small differences in activities such as cut and peel in food preparation. This is in contrast to traditional action recognition where the goal is to recognise full-body activities such as walking or jumping.

Chapter 3

Background Theory

You gain strength, courage and confidence by every experience in which you really stop to look fear in the face. You are able to say to yourself, 'I have lived through this horror. I can take the next thing that comes along.' You must do the thing you think you cannot do.

Eleanor Roosevelt

This chapter provides an overview of the relevant theory used for action analysis in the following chapters. We first describe the video descriptors. More specifically, a video is represented as a set of features extracted on a pixel basis. We also describe how we use this set of features to obtain both Riemannian features: *(i)* covariance features that lie in the space of Symmetric Positive Definite (SPD) matrices, and *(ii)* Linear Subspaces (LS) that lie in the space of Grassmann manifolds. We then summarise learning methods based on Gaussian Mixture Models (GMM), the Fisher vector (FV) representation as well as Riemannian manifolds.

3.1 Video Descriptors

Here, we describe how to extract from a video a set of features in a pixel level. The low-level video descriptor is the same for all the methods examined in this work.

3.1.1 Low-level Descriptors

A video $\mathcal{V} = \{\boldsymbol{I}_t\}_{t=1}^{T}$ is an ordered set of T frames. Each frame $\boldsymbol{I}_t \in \mathbb{R}^{r \times c}$ can be represented by a set of feature vectors $F_t = \{\boldsymbol{f}_p\}_{p=1}^{N_t}$. We extract the following $d = 14$ dimensional feature vector for each pixel in a given frame t [138]:

$$\boldsymbol{f} = [\, x,\ y,\ \boldsymbol{g},\ \boldsymbol{o}\,]^{\top} \tag{3.1}$$

where x and y are the pixel coordinates, while $\boldsymbol{g}$ and $\boldsymbol{o}$ are defined as:

$$\boldsymbol{g} = \left[\; |J_x|,\; |J_y|,\; |J_{yy}|,\; |J_{xx}|,\; \sqrt{J_x^2 + J_y^2},\; \text{atan}\frac{|J_y|}{|J_x|} \; \right] \tag{3.2}$$

$$\boldsymbol{o} = \left[\; u,\;\; v,\;\; \frac{\partial u}{\partial t},\;\; \frac{\partial v}{\partial t},\;\; \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right),\;\; \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right) \; \right] \tag{3.3}$$

The first four gradient-based features in Eq. (3.2) represent the first and second order intensity gradients at pixel location (x, y). The last two gradient features represent gradient magnitude and gradient orientation. The optical flow based features in Eq. (3.3) represent: the horizontal and vertical components of the flow vector, the first order derivatives with respect to time, the divergence and vorticity of optical flow [10], respectively.

Typically only a subset of the pixels in a frame correspond to the object of interest ($N_t < r \times c$). As such, we are only interested in pixels with a gradient magnitude greater than a threshold τ [54]. We discard feature vectors from locations with a small magnitude, resulting in a variable number of feature vectors per frame.

For each video $\mathcal{V}$, the feature vectors are pooled into set $\mathcal{F} = \{\boldsymbol{f}_n\}_{n=1}^{N}$ containing N vectors. This pooled set of features $\mathcal{F}$ can be used directly by methods such as GMM and FV. Describing these features using a Riemannian Manifold setting requires a further step to produce either a covariance matrix feature or a linear subspace feature.

3.1.2 Covariance Matrices of Features

A valid covariance matrix is also a Symmetric Positive Definite (SPD) matrix, and hence can be interpreted as a point on a Riemannian Manifold [9]. Covariance matrices of features have proved useful for action recognition [44, 54, 139]. The empirical estimate of the covariance matrix of set $\mathcal{F}$ is given by

$$\boldsymbol{C} = \frac{1}{N} \sum_{n=1}^{N} \left(\boldsymbol{f}_n - \overline{\mathcal{F}}\right) \left(\boldsymbol{f}_n - \overline{\mathcal{F}}\right)^\top \tag{3.4}$$

where $\overline{\mathcal{F}} = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{f}_n$ is the mean feature vector.

3.1.3 Linear Subspaces

The pooled feature vectors set $\mathcal{F}$ can be represented as a subspace through any orthogonalisation procedure like singular value decomposition (SVD) [58]. Let $\boldsymbol{F} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top$ be the SVD of $\mathcal{F}$. The first m columns of $\boldsymbol{U}$ represent an optimised subspace of order m. The Grassmann manifold $\mathcal{G}_{d,m}$ is the set of m-dimensional linear subspaces of $\mathbb{R}^d$. An element of $\mathcal{G}_{d,m}$ can be represented by an orthonormal matrix $\boldsymbol{Y}$ of size $d \times m$ such that $\boldsymbol{Y}^\top \boldsymbol{Y} = \boldsymbol{I}_m$, where $\boldsymbol{I}_m$ is the $m \times m$ identity matrix.

3.2 Gaussian Mixture Model

A GMM is a weighted sum of K component Gaussian densities [18], defined as:

$$p(\boldsymbol{f}|\lambda) = \sum_{k=1}^{K} w_k \mathcal{N}(\boldsymbol{f}|, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{3.5}$$

where $\boldsymbol{f}$ is a d-dimensional feature vector, w_k is the weight of the k-th Gaussian (with constrains $0 \leq w_k \leq 1$ and $\sum_{k=1}^{K} w_k = 1$), and $\mathcal{N}(\boldsymbol{f}|, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ is the component Gaussian density with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, given by:

$$\mathcal{N}(\boldsymbol{f}|, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{\frac{d}{2}} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp\left\{ -\frac{1}{2} (\boldsymbol{f} - \boldsymbol{\mu})^{\top} \boldsymbol{\Sigma}^{-1} (\boldsymbol{f} - \boldsymbol{\mu}) \right\}$$

The complete Gaussian mixture model is parameterised by the mean vectors, covariance matrices and weights of all component densities. These parameters are collectively represented by the notation:

$$\lambda = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\} \qquad k = 1, \cdots, K \tag{3.6}$$

There are two variants on the GMM. The covariance matrices $\boldsymbol{\Sigma}_k$ can be full rank or constrained to be diagonal [131]. The most popular method to estimate the parameters of the GMM (λ) is maximum likelihood (ML) estimation. Given the training data, the objective of ML estimation is to find the model parameters that maximise the likelihood of the GMM. For a sequence of T training vectors $\mathcal{X} = \{\boldsymbol{f}_1, \cdots, \boldsymbol{f}_T\}$, the GMM likelihood can be written as:

$$p(\mathcal{X}|\lambda) = \prod_{t=1}^{T} p(\boldsymbol{f}_t|\lambda). \tag{3.7}$$

As stated in [131], Eq. (3.7) is a non-linear function of the parameters λ and direct maximisation is not possible. However, a special case of the Expectation-maximisation (EM) algorithm can be used to iteratively estimate the ML parameters. The EM algorithm starts with an initial mode, λ, which is usually obtained by using the K-means algorithm. A new model $\overline{\lambda}$ is then estimated, such that $p(\mathcal{X}|\overline{\lambda}) > p(\mathcal{X}|\lambda)$. The new model then becomes the initial model for the next iteration and the process is repeated until some convergence threshold is reached [131].

For action classification, we learnt one GMM per action. This results in a set of GMM models that we will express as $\{\lambda_a\}_{a=1}^{A}$, where A is the total number of actions. For each testing video $\mathcal{V}$, the feature vectors in set $\mathcal{F}$ are assumed independent, so the average log-likelihood of a model λ_a is computed as:

$$\log p(\mathcal{F}|\lambda_a) = \frac{1}{N} \sum_{n=1}^{N} \log p(\boldsymbol{f}_n|\lambda_a) \tag{3.8}$$

We classify each video to the model a which has the highest average log-likelihood (Bayes' theorem).

3.3 Fisher Vector Representation

The FV approach encodes the deviations from a probabilistic visual dictionary, which is typically a GMM with diagonal covariance matrices [135]. The parameters of a GMM with K components can be expressed as $\lambda = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\sigma}_k\}_{k=1}^{K}$, where, w_k is the weight, $\boldsymbol{\mu}_k$ is the mean vector, and $\boldsymbol{\sigma}_k$ is the diagonal covariance matrix for the k-th Gaussian. The parameters are learnt using the Expectation Maximisation algorithm [18] on training data.

Given the pooled set of features $\mathcal{F}$ from video $\mathcal{V}$, the deviations from the GMM are then accumulated using [135]:

$$\mathcal{G}_{\boldsymbol{\mu}_k}^{\mathcal{F}} = \frac{1}{N\sqrt{w_k}} \sum_{n=1}^{N} \gamma_n(k) \left(\frac{\boldsymbol{f}_n - \boldsymbol{\mu}_k}{\boldsymbol{\sigma}_k} \right) \tag{3.9}$$

$$\mathcal{G}_{\boldsymbol{\sigma}_k}^{\mathcal{F}} = \frac{1}{N\sqrt{2w_k}} \sum_{n=1}^{N} \gamma_n(k) \left[\frac{(\boldsymbol{f}_n - \boldsymbol{\mu_k})^2}{\boldsymbol{\sigma}_k^2} - 1 \right] \tag{3.10}$$

where vector division indicates element-wise division and $\gamma_n(k)$ is the posterior probability of $\boldsymbol{f}_n$ for the k-th component:

$$\gamma_n(k) = \frac{w_k \mathcal{N}(\boldsymbol{f}_n | \boldsymbol{\mu}_k, \boldsymbol{\sigma}_k)}{\sum_{i=1}^{K} w_i \mathcal{N}(\boldsymbol{f}_n | \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i)} \tag{3.11}$$

The Fisher vector for each video $\mathcal{V}$ is represented as the concatenation of $\mathcal{G}_{\boldsymbol{\mu}_k}^{\mathcal{F}}$ and $\mathcal{G}_{\boldsymbol{\sigma}_k}^{\mathcal{F}}$ (for $k = 1, \ldots, K$) into vector $\mathcal{G}_{\lambda}^{\mathcal{F}}$. As $\mathcal{G}_{\boldsymbol{\mu}_k}^{\mathcal{F}}$ and $\mathcal{G}_{\boldsymbol{\sigma}_k}^{\mathcal{F}}$ are d-dimensional, $\mathcal{G}_{\lambda}^{\mathcal{F}}$ has the dimensionality of $2dK$. Power normalisation is then applied to each dimension in $\mathcal{G}_{\lambda}^{\mathcal{F}}$. The power normalisation to improve the FV for classification was proposed in [120] of the form $z \leftarrow \text{sign}(z)|z|^{\rho}$, where z corresponds to each dimension and the power coefficient $\rho = 1/2$.

Finally, l_2-normalisation is applied. Note that the deviations for the weights are usually omitted as they add little information [135]. The FVs are fed to a linear SVM for classification, where the similarity between vectors is measured using dot-products [135].

3.4 Classification on Riemannian Manifolds

In this section, we briefly describe several approaches for classification on Riemannian manifolds including Nearest-Neighbour, Euclidean-mapping methods using kernels, and kernelised sparse representations.

3.4.1 Nearest-Neighbour Classifier

The Nearest Neighbour (NN) approach classifies a query data based on the most similar observation in the annotated training set [105]. To decide whether two observations are similar we will employ two metrics: the log-Euclidean distance for SPD matrices [54] and the Projection Metric for LS [56].

The log-Euclidean distance (d_{spd}) is one of the most popular metrics for SPD matrices due to its accuracy and low computational complexity [13]. Formally, it is defined as:

$$d_{\text{spd}}(\boldsymbol{C}_1, \boldsymbol{C}_2) = ||\log(\boldsymbol{C}_1) - \log(\boldsymbol{C}_2)||_F^2 \tag{3.12}$$

where $\log(\cdot)$ is the matrix-logarithm and $||\cdot||_F$ denotes the Frobenius norm on matrices.

As for LS, a common metric to measure the similarity between two subspaces is via principal angles [56]. The metric can include the smallest principal angle, the largest principal angle, or a combination of all principal angles [56, 161]. In this work we have selected the Projection Metric which uses all the principal angles [56]:

$$d_{\text{ls}}(\boldsymbol{Y}_1, \boldsymbol{Y}_2) = \left(m - \sum_{i=1}^{m} \cos^2 \theta_i \right)^{1/2} \tag{3.13}$$

where m is the size of the subspace. The principal angles can be easily computed from the SVD of

$$\boldsymbol{Y}_1^\top \boldsymbol{Y}_2 = \boldsymbol{U}(\cos\Theta)\boldsymbol{V}^\top \tag{3.14}$$

where $\boldsymbol{U} = [\boldsymbol{u}_1, \cdots, \boldsymbol{u}_m]$, $\boldsymbol{V} = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_m]$, and $\cos\Theta = \text{diag}(\cos\theta_1, \cdots, \cos\theta_m)$.

3.4.2 Kernel Approach

Manifolds can be mapped to Euclidean spaces using Mercer kernels [182]. Kernel functions transform the data into a higher dimensional space which affords to perform linear separation. This mapping transformation allows us to employ algorithms originally formulated for $\mathbb{R}^n$ with manifold value data [67, 161]. The kernels are used in combination with Support Vector Machines (SVMs).

Kernels for SPD matrices. Several kernels for the set of SPD matrices have been proposed in the literature [67, 173, 182]. One kernel based on the log-Euclidean distance is derived in [173] and various kernels can be generated, among them [161]:

$$K_{\text{spd}}^{\text{rbf}}(\boldsymbol{C}_1, \boldsymbol{C}_2) = \exp\left(-\gamma_r \cdot ||\log(\boldsymbol{C}_1) - \log(\boldsymbol{C}_2)||_F^2\right) \tag{3.15}$$

$$K_{\text{spd}}^{\text{poly}}(\boldsymbol{C}_1, \boldsymbol{C}_2) = \left(\gamma_p \cdot \text{tr}\left[\log(\boldsymbol{C}_1)^\top \log(\boldsymbol{C}_2)\right]\right)^d \tag{3.16}$$

where *poly* and *rbf* stands for polynomial and radial basic functions, respectively.

Kernels for LS. Similar to SPD kernels, many kernels have been proposed for LS [58, 145, 182]. Various kernels can be generated from the projection metric, among them [161]:

$$K_{\text{ls}}^{\text{rbf}}(\boldsymbol{Y}_1, \boldsymbol{Y}_2) = \exp\left(-\gamma_r \cdot ||\boldsymbol{Y}_1\boldsymbol{Y}_1^\top - \boldsymbol{Y}_2\boldsymbol{Y}_2^\top||_F^2\right) \tag{3.17}$$

$$K_{\text{ls}}^{\text{poly}}(\boldsymbol{Y}_1, \boldsymbol{Y}_2) = \left(\gamma_p \cdot ||\boldsymbol{Y}_1^\top\boldsymbol{Y}_2||_F^2\right)^m \tag{3.18}$$

The parameters γ_r and γ_p are the kernel parameters.

3.4.3 Kernelised Sparse Representation

Recently, several works show the efficacy of sparse representation methods for addressing manifold feature classification problems [178, 57]. Here, each manifold point is represented by its sparse coefficients.

Let $\mathcal{X} = \{\boldsymbol{X}_j\}_{j=1}^J$ be a population of Riemannian points (where $\boldsymbol{X}_j$ is either a SPD matrix or a LS) and $\mathcal{D} = \{\boldsymbol{D}_i\}_{i=1}^K$ be the Riemannian dictionary of size K, where each element represents an atom. Given a kernel $k(\cdot,\cdot)$, induced by the feature mapping function $\phi : \mathbb{R}^d \to \mathbb{H}$, we seek to learn a dictionary and corresponding sparse code $\boldsymbol{s} \in \mathbb{R}^K$ such that $\phi(\boldsymbol{X})$ can be well approximated by the dictionary $\phi(\mathcal{D})$.

The kernelised dictionary learning in Riemannian manifolds optimises the following objective function [178, 57]:

$$\min_{\boldsymbol{s}} \left(||\phi(\boldsymbol{X}) - \sum_{i=1}^K s_i\phi(\boldsymbol{D}_i)||_F^2 + \lambda||\boldsymbol{s}||_1 \right) \tag{3.19}$$

over the dictionary and the sparse codes $\mathcal{S} = \{\boldsymbol{s}_j\}_{j=1}^J$. After initialising the dictionary $\mathcal{D}$, the optimisation function is solved by repeating two steps (sparse coding and dictionary update). In the sparse coding step, $\mathcal{D}$ is fixed and $\mathcal{S}$ is computed. In the dictionary update step, $\mathcal{S}$ is fixed while $\mathcal{D}$ is updated, with each dictionary atom updated independently.

For the sparse representation on SPD matrices, each atom $\boldsymbol{D}_i \in \mathbb{R}^{d\times d}$ and each element $\boldsymbol{X} \in \mathbb{R}^{d\times d}$ are SPD matrices. The dictionary is learnt following [59], where the dictionary is initialised using the Karcher mean [17]. For the sparse representation on LS, the dictionary $\boldsymbol{D}_i \in \mathbb{R}^{d\times m}$ and each element $\boldsymbol{X} \in \mathbb{R}^{d\times m}$ are elements of $\mathcal{G}_{d,m}$ and need to be determined by the Kernelised Grassmann Dictionary Learning algorithm proposed in [57]. We refer to the kernelised sparse representation (KSR) for SPD matrices and LS as KSR_{spd} and KSR_{ls}, respectively.

Chapter 4

Datasets for Action Recognition

One of the lessons that I grew up with was to always stay true to yourself and never let what somebody else says distract you from your goals. And so when I hear about negative and false attacks, I really don't invest any energy in them, because I know who I am.

Michelle Obama

This chapter overviews the existing datasets used for action recognition and also action recognition and segmentation in this thesis. The datasets under consideration are: KTH, UCF-Sports, UT-Tower, and CMU-MMAC.

This chapter describes the scenario were the dataset were recorded, the number of videos, number of subjects, among other important facts. This chapter also provides examples of the actions collected per each dataset.

4.1 KTH

The KTH dataset [140] contains 25 subjects performing 6 types of human actions and 4 scenarios. The actions included in this dataset are: boxing, handclapping, handwaving, jogging, running, and walking. The scenarios include indoor, outdoor, scale variations, and varying clothes. See Fig. 4.1 for examples of actions and scenarios, where each row is a different scenario. Each original video of the KTH dataset contains an individual performing the same action.

Each action is performed four times and each subdivision or action-instance (in terms of start-frame and end-frame) is provided as part of the dataset. This dataset contains 2391 action-instances, with a length between 1 and 14 seconds [14]. The average length is four seconds. The sequences were downsampled to the spatial resolution of 160×120 pixels, and temporal resolution is 25 frames per second.

Figure 4.1: The KTH dataset contains 6 actions performed by 25 subjects. Each row is a different scenario.

4.2 UCF-Sports

The UCF-Sports dataset [133] is a collections of 150 sport videos or sequences in unconstrained environments with the resolution of 720×480. This dataset consists of a set of actions collected from various sports which are typically featured on broadcast television channels such as the BBC and ESPN.

This datasets consists of the following 10 actions: diving, golf swinging, kicking a ball, lifting weights, riding horse, running, skate boarding, pommel horse, high bar, and walking. See Fig. 4.2 for examples of actions in this dataset. The number of videos per action varies from 6 to 22. The videos presented in this dataset have varying backgrounds, a wide range of scenes, and viewpoints. The bounding box enclosing the person of interest is provided with the dataset for the majority of the videos.

4.3 UT-Tower

The UT-Tower dataset [31] contains 9 actions performed 12 times. See Fig. 4.3 for examples of actions in this dataset. In total, there are 108 low-resolution videos. Low-resolution video is the usual scenario when recorded from a distant view, and is the common setting for military and surveillance applications [31]. The average height of human figures in this dataset is about 20 pixels. Additionally, shadows and blurry videos make this dataset more complex. The actions include: pointing, standing,

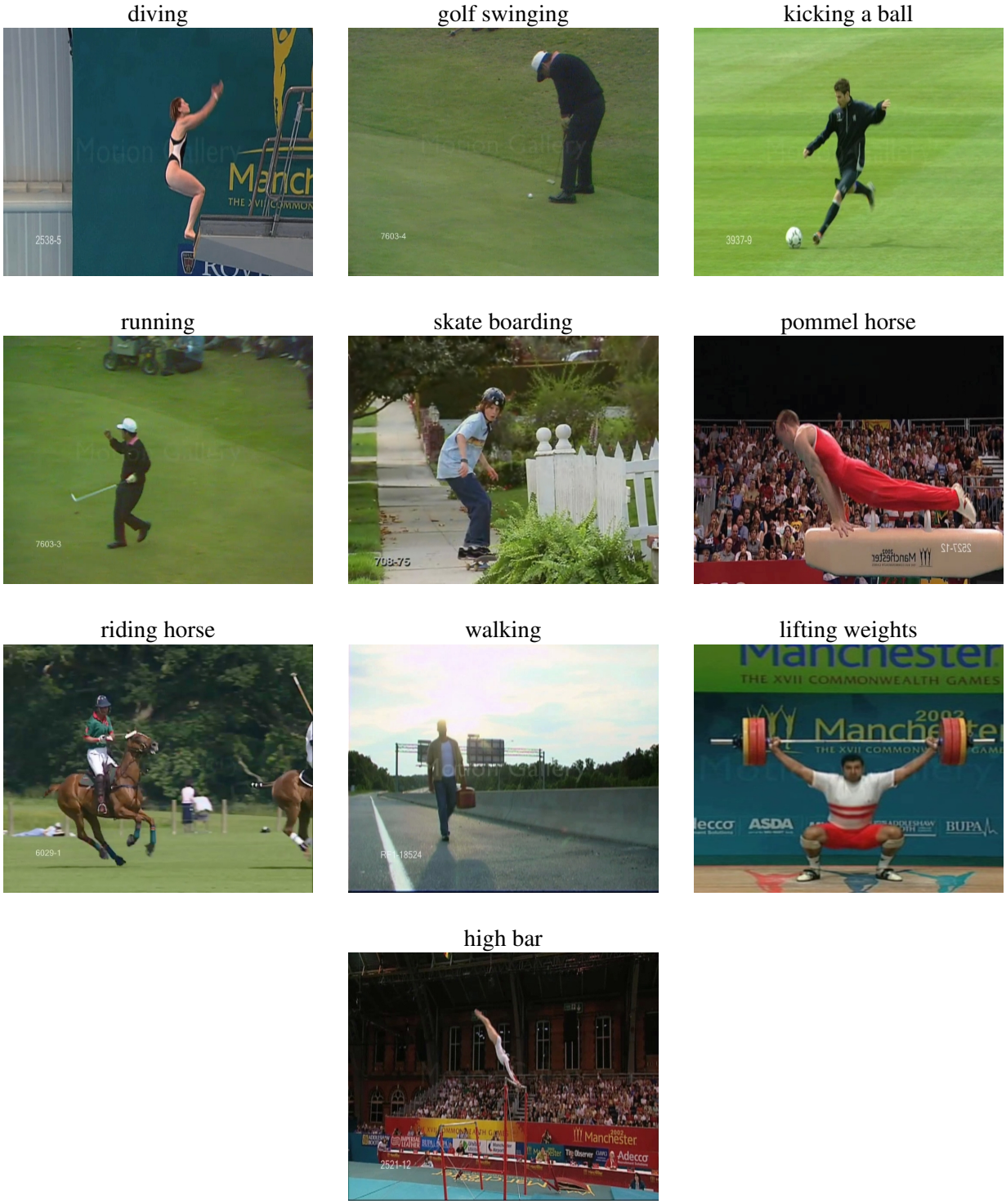


Figure 4.2: The UCF dataset contains 10 actions collected in unconstrained environments.

digging, walking, carrying, running, wave1, wave2, and jumping. The videos were recorded in two scenarios: concrete square and lawn. In the concrete square scenario the actions recorded are: pointing, standing, digging, and walking. In the lawn scenario the actions recorded are: carrying, running, wave with one hand, wave both hands, and jumping. In both cases, the camera is stationary with jitter and the temporal resolution is 10 frames per second. This dataset also provides the bounding boxes.

Figure 4.3: The UT-Tower dataset contains 9 actions. All videos have low resolution.

4.4 CMU-MMAC

The CMU Multi-Modal Activity Database (CMU-MMAC) database contains multi-modal measures (audio, video, accelerations, motion capture) of the human activity of subjects performing the tasks involved in cooking and food preparation. The CMU-MMAC database was collected in Carnegie Mellon's Motion Capture Lab. A kitchen was built and twenty-five subjects have been recorded cooking based on five recipes: brownies, pizza, sandwich, salad, and scrambled eggs [40].

For example, the activity making brownies from a dry mix box has video clips of twelve individuals preparing brownies in a spontaneous way, without receiving instructions on how to perform each task. Each video depicts a person performing a sequence of actions, with each action belonging to one of 14 classes such as pouring, spraying, and stirring. See Fig. 4.4 for examples of actions for the activity making brownies.

This dataset is constrained to a single context and is recorded in a laboratory under well controlled conditions using five external cameras and one wearable camera. Fig 4.5b shows the views from external and wearable cameras. 3D skeletal computed from external cameras is depicted in Fig 4.5a.

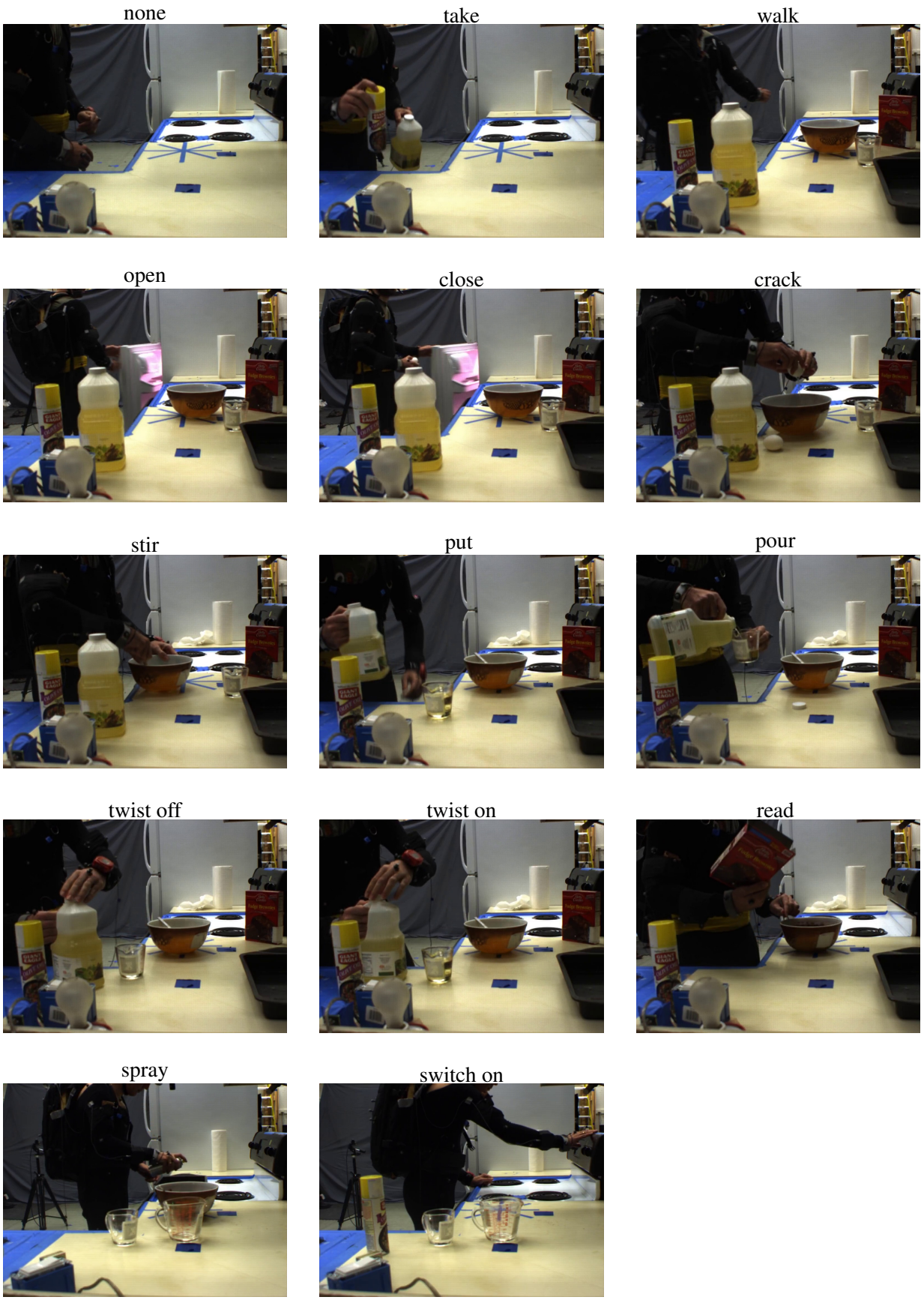
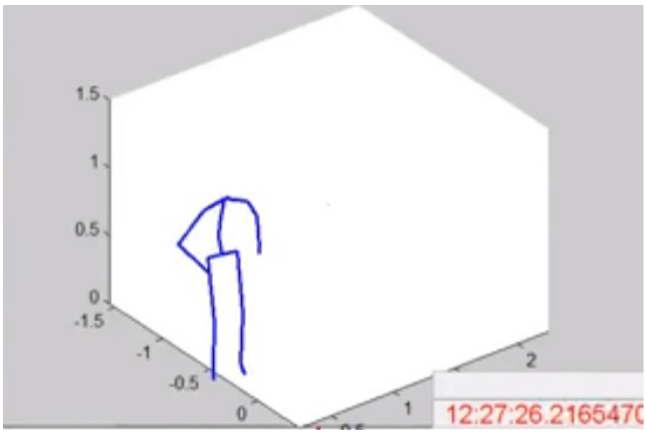


Figure 4.4: The CMU-MMAC dataset records 5 cooking recipes. Examples of the 14 actions included in the recipe for brownies.



(a) 3D Human Skeletal.

(b) Views from external and wearable cameras.

Figure 4.5: CMU-MMAC dataset.

Chapter 5

Comparative Evaluation of Action Recognition Techniques

We ourselves feel that what we are doing is just a drop in the ocean. But the ocean would be less because of that missing drop.

Mother Teresa

This chapter[1] presents a comparative evaluation of various techniques for action recognition while keeping as many variables as possible controlled. Riemannian manifolds have been recently explored for the problem of action recognition. Two categories of Riemannian manifolds are employed: symmetric positive matrices and linear subspaces. For both categories their corresponding nearest neighbour classifiers, kernels, and recent kernelised sparse representations are used. This chapter also compares against traditional action recognition techniques based on Gaussian mixture models (GMMs) and Fisher vectors (FVs). These action recognition techniques are evaluated under ideal conditions, as well as their sensitivity in more challenging conditions (variations in scale and translation). Despite recent advancements for handling manifolds, manifold based techniques obtain the lowest performance and their kernel representations are more unstable in the presence of challenging conditions. The FV approach obtains the highest accuracy under ideal conditions. Moreover, FV best deals with moderate scale and translation changes.

5.1 Introduction

Recently, there has been an increasing interest on action recognition using Riemannian manifolds. Such recognition systems can be roughly placed into two main categories: **(i)** based on linear subspaces (LS), and **(ii)** based on symmetric positive definite (SPD) matrices. The space of m-dimensional

[1] The work presented in this chapter has been published in [27].

LS in $\mathbb{R}^n$ can be viewed as a special case of Riemannian manifolds, known as Grassmann manifolds [158].

Other techniques have been also applied for the action recognition problem such as GMMs, bag of visual words (BoVW), and FVs. In [26, 92] each action is represented by a combination of GMMs and then the decision making is based on the principle of selecting the most probable action according to Bayes' theorem [154]. The FV representation can be thought as an evolution of the BoVW representation, encoding additional information [36]. Rather than encoding the frequency of the descriptors for a given video, FV encodes the deviations from a probabilistic version of the visual dictionary (which is typically a GMM) [168].

Several review papers have compared various techniques for human action recognition. The majority of those reviews have focused on the taxonomy classification of existing techniques [6, 78, 124, 175]. An alternative approach to review the action recognition problem has been embraced in [61]. Instead of reviewing the taxonomy of the benchmarks based on to their architecture and functions, [61] provides a review of what computer vision systems can and cannot do, by considering the datasets used to test them.

The aforementioned reviews describe existing techniques for human action recognition, show how this research area has progressed throughout the years, and the current advantages and limitations of the state-of-the-art are discussed. They also provide some directions of research or how these limitations can be addressed. However, none of them focus on how well various action recognition systems work across same datasets *and* same extracted features.

An earlier comparison of classifiers for human activity recognition is studied [119]. The performance comparison with seven classifiers in one single dataset is reported. Although this work presents a broad range of classifiers, it fails to provide a more extensive comparison by using more datasets and hence its conclusions may not generalise to other datasets.

So far there has been no systematic comparison of performance between methods based on SPD matrices and LS. Furthermore, there has been no comparison of manifold based methods against traditional action recognition methods based on GMMs and FVs in the presence of realistic and challenging conditions. Lastly, existing review papers fail to compare various classifiers using the same features across several datasets.

To address the aforementioned problems, this chapter provides a more detailed analysis of the performance of the aforementioned methods under the same set of features. To this end, three popular datasets are used: KTH [140], UCF-Sports [133] and UT-Tower [31]. Nearest-neighbour classifiers, kernels as well as recent kernelised sparse representations are used for the Riemannian representations. Finally, it is also quantitatively shown when these methods break and how the performance degrades when the datasets have challenging conditions (translations and scale variations).

All the background theory needed for this chapter is described in Chapter 3. For a fair comparison across all approaches, the same set of features are used, as explained in Section 3.1. More specifically, a video is represented as a set of features extracted on a pixel basis. Section 3.1 also describes how to use this set of features to obtain both Riemannian features: (1) covariance features that lie in the

space of SPD matrices, and (2) LS that lie in the space of Grassmann manifolds. Classification using GMM, FV, and Riemannian manifolds are explained in Sections 3.2, 3.3, and 3.4, respectively.

This chapter continues as follows. The datasets and experiment setup are described in Section 5.2. In Section 5.3, we present the experimental results with three datasets under ideal and challenging conditions. The main findings and future work are summarised in Section 5.4.

5.2 Datasets and Setup

For the experiments three datasets are used: KTH [140], UCF-Sports [133], and UT-Tower [31]. A detail description of all datasets is given in Chapter 4. For the experiments with the KTH dataset only scenario 1 is used.

The Leave-One-Out (LOO) protocol suggested by each datasets is employed. The LOO protocol takes out one sample video for testing and uses all of the remaining videos for training. This is executed for every testing video in a cycling basis, and the overall accuracy is attained by simply averaging the accuracy of all iterations [149]. For UT-Tower and UCF-Sports, one sample video is left out for testing on a rotating basis. For KTH one person is left out. For each video, a set of $d = 14$ dimensional features vectors are extracted (see Section 3.1). We only use feature vectors with a gradient magnitude greater that a threshold β. The threshold β used for selecting low-level feature vectors was set to 40 based on preliminary experiments.

For each video, we obtain one SPD matrix and one LS. In order to obtain the optimised linear subspace $\mathcal{G}_{d,m}$ in the the manifold representation, we vary $m = 1, \cdots d$. The approaches used for classification using Riemannian Manifolds are explained in Section 3.4. We test with manifolds kernels using various parameters. The set of parameters was used as proposed in [161]. Polynomial kernels $K_{\text{spd}}^{\text{poly}}$ and $K_{\text{ls}}^{\text{poly}}$ are generated by taking $\gamma_p = 1/d_p$ and $d_p = \{1, 2, \cdots, d\}$. Projection RBF kernels are generated with $\gamma_r = \frac{1}{d}2^{\delta}$ where $\delta = \{-10, -9, \cdots, 9\}$ for $K_{\text{spd}}^{\text{rbf}}$, and $\delta = \{-14, -12, \cdots, 20\}$ for $K_{\text{ls}}^{\text{rbf}}$. For the sparse representation of SPD matrices and LS we have used the code provided by [59] and [57], respectively. Kernels are used in combination with SVM for final classification. We report the best accuracy performance after iterating with various parameters.

For the FV representation, we use the same set-up as in [168]. We randomly sampled 256,000 features from training videos and then the visual dictionary is learnt with 256 Gaussians. Each video is represented by a FV. The FVs are fed to a linear SVM for classification. For the GMM modelling, we learn a model for each action using all the feature vectors belonging to the same action. For each action a GMM is trained with K=256 components.

5.3 Comparative Evaluation

We perform two sets of experiments: **(1)** in ideal conditions, where the classification is carried out using the original dataset (Section 5.3.1), and **(2)** in realistic and challenging conditions where testing videos are modified by scale changes and translations (Section 5.3.2).

5.3.1 Ideal Conditions

We start our experiments using the NN classifier for both Riemannian representations: SPD matrices and LS. For LS we employ the projection metric as per Eq. (3.13) and for SPD matrices we employ the log-Euclidean distance as per Eq. (3.12). We tune the parameter m (subspace order) for each dataset. The kernels selected for SPD matrices and LS are described in Eqs. (3.15)-(3.18) and their parameters are selected as explained in Section 5.2.

We present a summary of the best performance obtained for the manifold representations using the optimal subspace for LS and also the optimal kernel parameters for both representations. Similarly, we report the best accuracy performance for the kernelised sparse representations KSR_{spd} and KSR_{ls}. Moreover, we include the performance for the GMM and FV representations.

The results are presented in Table 5.1. First of all, we observe that using a SVM for action recognition usually leads to a better accuracy than NN. In particular, we notice that the NN approach performs quite poorly. The NN classifier may not be effective enough to capture the complexity of the human actions when there is insufficient representation of the actions (one video is represented by one SPD matrix or one LS). Secondly, we observe that among the manifold techniques, SPD based approaches perform better than LS based approaches. While LS capture only the dominant eigenvectors [144], SPD matrices capture both the eigenvectors and eigenvalues [156]. The eigenvalues of a covariance matrix typify the variance captured in the direction of each eigenvector [156].

Despite KSR_{spd} has shown superior performance in other computer vision tasks such as face recognition, texture classification, and person re-identification [59]; it is not the case for the action recogni-

Table 5.1: Accuracy of action recognition in ideal conditions.

	KTH	UCF-Sports	UT-Tower	average
d_{spd} + NN	76.0%	76.5%	73.1%	75.2%
d_{ls} + NN	67.3%	65.7%	76.8%	69.9%
$K_{\text{spd}}^{\text{poly}}$ + SVM	92.0%	75.2%	87.9%	<u>85.0%</u>
$K_{\text{spd}}^{\text{rbf}}$ + SVM	84.0%	79.2%	81.5%	81.6%
$K_{\text{ls}}^{\text{poly}}$ + SVM	56.0%	50.3%	42.6%	49.6%
$K_{\text{ls}}^{\text{rbf}}$ + SVM	76.0%	61.7%	79.6%	72.4%
KSR_{spd} + SVM	80.0%	76.5%	81.5%	79.3%
KSR_{ls} + SVM	74.0%	72.5%	83.3%	77.3%
GMM	86.7%	80.5%	87.9%	<u>85.0%</u>
FV + SVM	**96.7%**	**88.6%**	**92.5%**	<u>**92.6%**</u>

tion problem. We conjecture this is due to the lack of labelled training data (each video is represented by only one SPD matrix), which may yield a dictionary with bad generalisation power. Moreover, sparse representations can be over-pruned, being caused by discarding several representative points that may be potentially useful for prediction [62].

Although kernel approaches map the data into higher spaces to allow linear separability, K_{spd}^{poly} exhibits on average a similar accuracy to GMM which does not transform the data. GMM is a weighted sum of Gaussian probability densities, which in addition to the covariance matrices, it uses the means and weights to determine the average log-likelihood of a set of feature vectors from a video to belong to a specific action.

While SPD kernels only use covariance matrices, GMMs use both covariance matrices and means. The combination of both statistics has proved to increase the accuracy performance in other classification tasks [7, 135]. FV outperforms all the classification methods with an average accuracy of 92.6%, which is 7.6 points higher than both GMM and K_{spd}^{poly}.

Similarly to GMM, FV also incorporates first and second order statistics (means and covariances), but it has additional processing in the form of power normalisation. It is shown in [120] that when the number of Gaussians increases, the FV turns into a sparser representation and the it negatively affects the linear SVM which measures the similarity using dot-products. The power normalisation unsparsifies the FV making it more suitable for linear SVMs. The additional information provided by the means and the power normalisation explains the superior accuracy performance of FV over all the classifier analysed.

5.3.2 Challenging Conditions

For this set of experiments, the training is carried out using the original datasets. For the analysis of translations, we have translated (shifted) each testing video vertically and horizontally. For the evaluation under scale variations, each testing video is shrunk or magnified. For both cases, we replace the missing pixels simply by copying the nearest rows or columns. See Fig. 5.1 for examples of videos under challenging conditions.

In this section, we evaluate the performance on all datasets under consideration when the testing videos have translations and scale variations. We have selected the following approaches for this evaluation: d_{spd}, $K_{\text{spd}}^{\text{poly}}$, $K_{\text{ls}}^{\text{rbf}}$, and FV. We discard d_{ls}, as its performance is too low and presents similar behaviour to d_{spd}. We do not include experiments on KSR_{spd} and KSR_{ls}, as we found that they show similar trends as $K_{\text{spd}}^{\text{poly}}$ and $K_{\text{ls}}^{\text{rbf}}$, respectively. Alike, GMM exhibits similar behaviour as FV.

The results for scale variations and translations are shown in Figs. 5.2 and Figs. 5.3, respectively. These results reveal that all the analysed approaches are susceptible to translation and scale variations. Both kernel based methods, $K_{\text{spd}}^{\text{poly}}$ and $K_{\text{ls}}^{\text{rbf}}$, exhibit sharp performance degradation even when the scale is only magnified or compressed by a factor of 0.05%. Similarly, for both kernels the accuracy rapidly decreases with a small translation. The NN classification using the log-Euclidean distance (d_{spd}) is less sensitive to both variations. It can be explained by the fact that log-Euclidean metrics are by definition invariant by any translation and scaling in the domain of logarithms [13]. FV presents

the best behaviour under moderate variations in both scale and translation. We attribute this to the loss of explicit spatial relations between object parts.

5.4 Conclusions

In this chapter, we have presented an extensive empirical comparison among existing techniques for the human action recognition problem. We have carried out our experiments using three popular datasets: KTH, UCF-Sports and UT-Tower. We have analysed Riemannian representations including

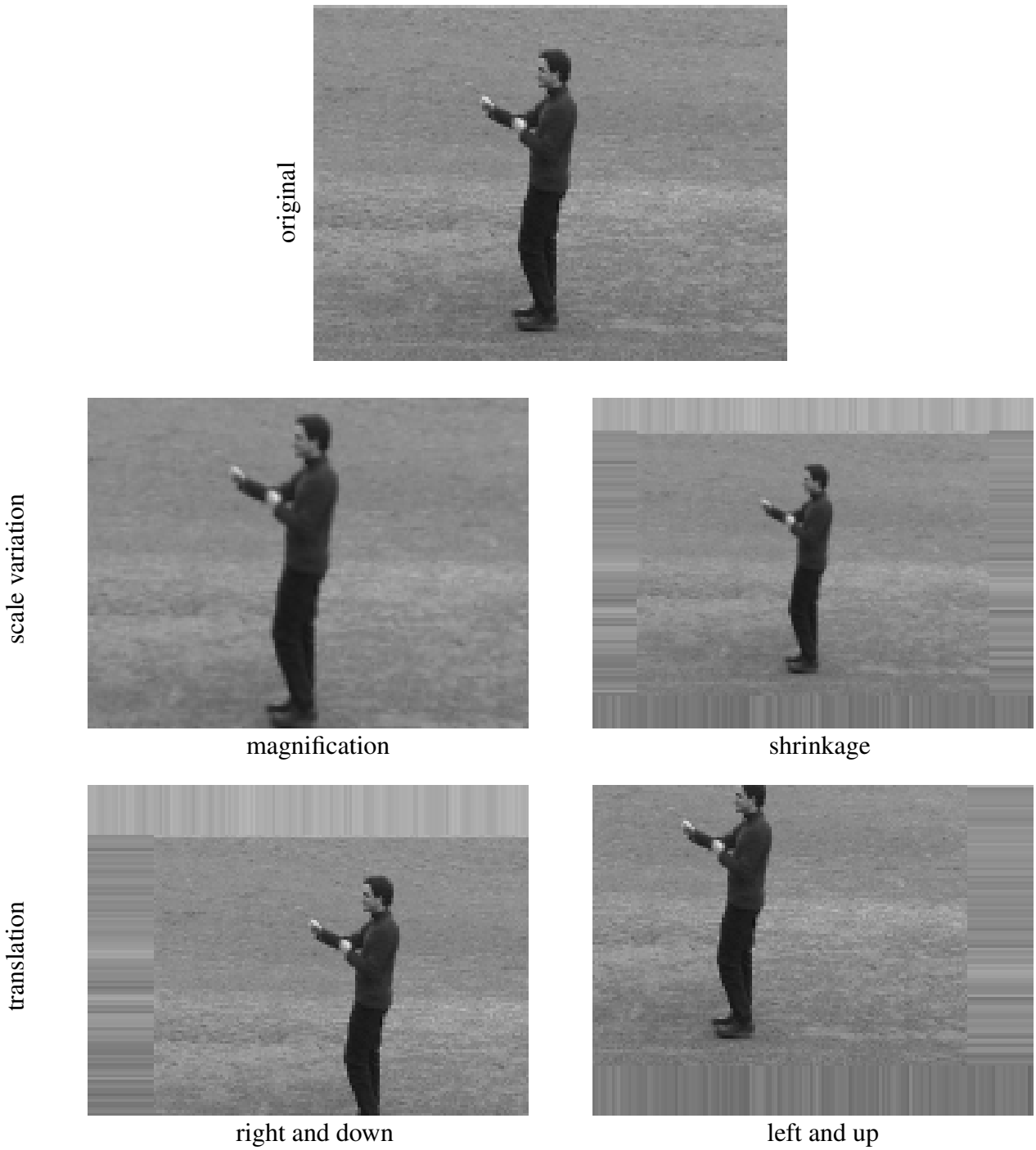


Figure 5.1: Examples of challenging conditions.

nearest-neighbour classification, kernel methods, and kernelised sparse representations. For Riemannian representation we used covariance matrices of features, which are symmetric positive definite (SPD), as well as linear subspaces (LS). Moreover, we compared all the aforementioned Riemannian representations with GMM and FV based representations, using the same extracted features. We also evaluated the robustness of the most representative approaches to translation and scale variations.

For manifold representations, all SPD matrices approaches surpass their LS counterpart, as a result of the use of not only the dominant eigenvectors but also the eigenvalues. The FV representation outperforms all the techniques under ideal and challenging conditions. Under ideal conditions, FV achieves an overall accuracy of 92.6%, which is 7.6 points higher than both GMM and the polynomial kernel using SPD matrices (K^{poly}_{spd}). FV encodes more information than Riemmannian based methods, as it characterises the deviation from a probabilistic visual dictionary (a GMM) using means and covariance matrices. Moreover, FV is less sensitive under moderate variations in both scale and translation.

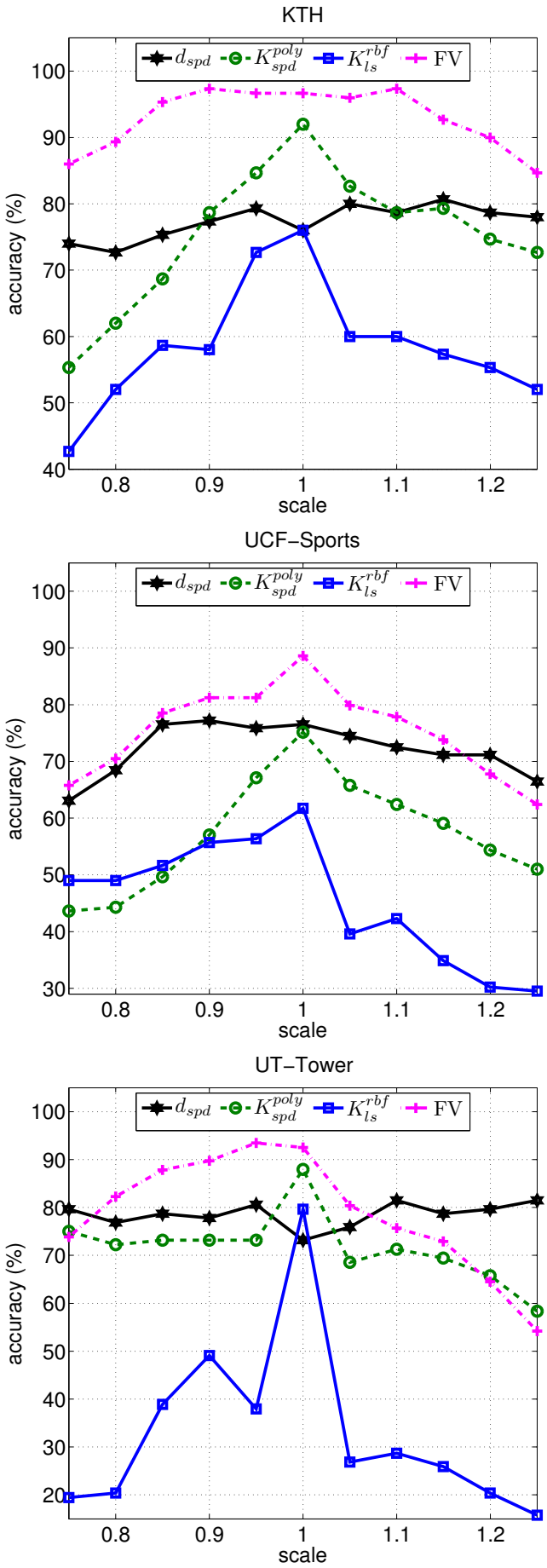


Figure 5.2: Scale variation results for all datasets. Scale variation above one means magnification, while below one means shrinking.

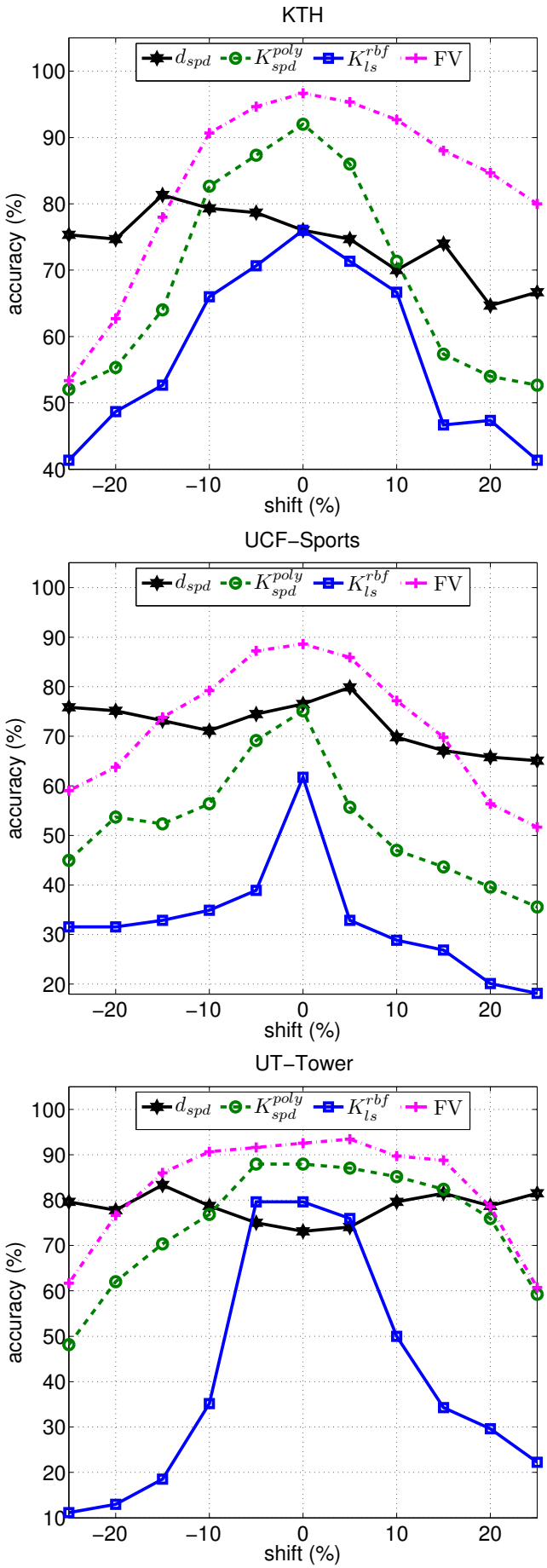


Figure 5.3: Translation results for all datasets. Each testing video is translated vertically and horizontally at the same time. A positive percentage indicates the video has been translated to the right and bottom while than a negative percentage indicates the video has been translated to the left and up.

Chapter 6

Joint Recognition and Segmentation of Actions

We can each define ambition and progress for ourselves. The goal is to work toward a world where expectations are not set by the stereotypes that hold us back, but by our personal passion, talents and interests.

Sheryl Sandberg

This chapter[1] presents two hierarchical approaches that perform joint classification and segmentation. For the first approach, a given video (containing several consecutive actions) is processed via a sequence of overlapping temporal windows. Each frame in a temporal window is represented through selective low-level spatio-temporal features which efficiently capture relevant local dynamics. Features from each window are represented as a Fisher vector, which captures first and second order statistics. Instead of directly classifying each Fisher vector, it is converted into a vector of class probabilities. The second proposed approach is based on Gaussian Mixture Models (GMMs). This GMM approach also processes a given video via a sequence of overlapping temporal windows. The vector of class probabilities for the GMM approach is obtained using the average log-likelihood over each temporal window. For both proposed approaches, the final classification decision for each frame is then obtained by integrating the class probabilities at the frame level, which exploits the overlapping of the temporal windows. Experiments were performed on two datasets: s-KTH (a stitched version of the KTH dataset to simulate consecutive single actions), and the challenging CMU-MMAC dataset. On s-KTH, the proposed approaches achieve an accuracy of 85.0% and 78.3% for the FV approach and the GMM approach, respectively. Both proposed methods significantly outperform one recent approach based on HMMs which obtained 71.2%. On CMU-MMAC, the proposed approach based on FV achieves an accuracy of 40.9%, outperforming the HMM approach which obtained 38.4%.

[1] The work presented in this chapter was first published in [26] and the extension was published in [24].

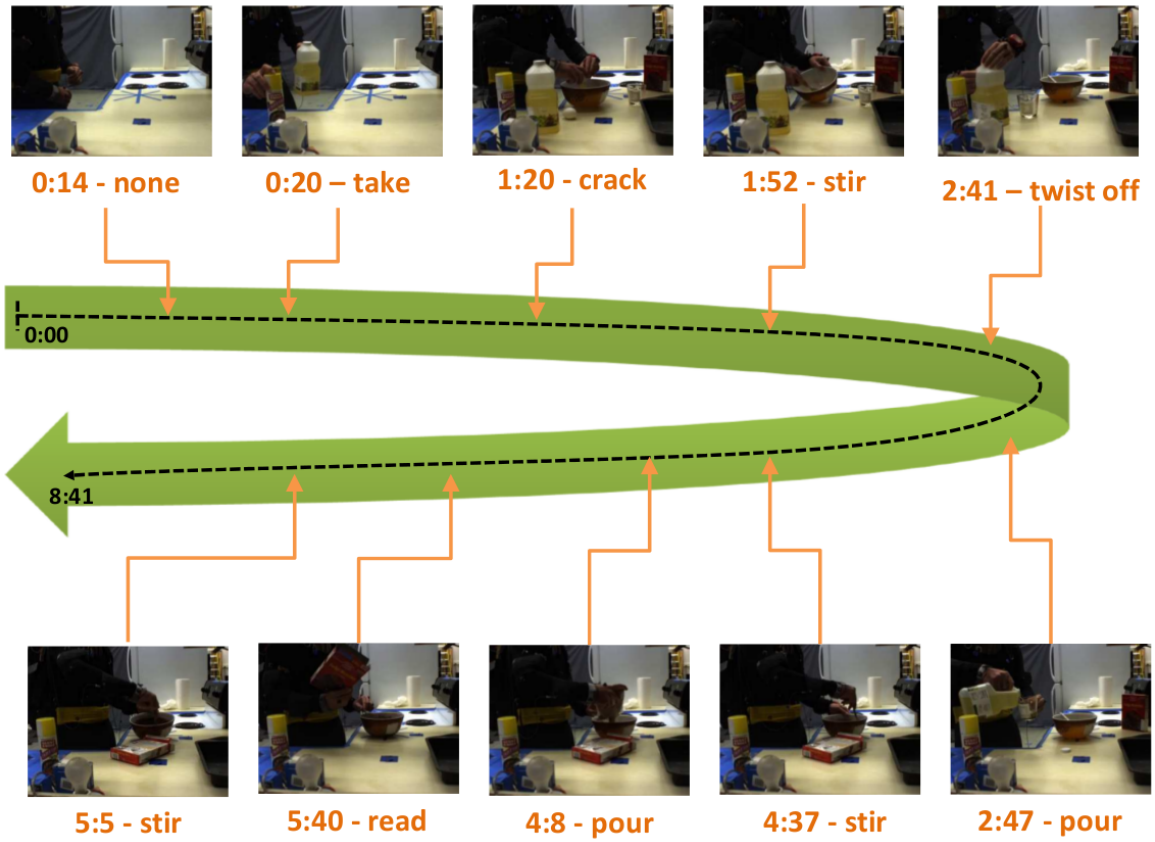


Figure 6.1: Example of a video with a sequence of several actions. The task is to correctly segment and recognise the actions presented in the sequence.

6.1 Introduction

In most computer vision literature, action recognition approaches have concentrated on single actions, where each video to be classified contains only one action. However, when observing realistic human behaviour in natural settings, the fundamental problem is segmenting and recognising actions from a sequence containing several single actions [21]. See Fig. 6.1 for an example of a video containing a sequence of several actions. It is challenging due to the high variability of appearances, shapes, possible occlusions, large variability in the temporal scale and periodicity of human actions, the complexity of articulated motion, the exponential nature of all possible movement combinations, as well as the prevalence of irrelevant background [63, 142].

Hoai et al. [63] address joint segmentation and classification by classifying temporal regions using a multi-class Support Vector Machine (SVM) and performing segmentation using dynamic programming. A similar approach is presented in [33], where the temporal relationship between actions is considered. Borzeshi et al. [19] proposed the use of hidden Markov models (HMMs) with multiple irregular observations (termed HMM-MIO) to perform action recognition and segmentation. A drawback of [19, 63, 33] is that they have a large number of parameters to optimise. Furthermore, [19] requires an extra stage to reduce dimensionality due to use of very high dimensional feature vectors, while [63, 33] require fully labelled annotations for training.

Typically, the aforementioned approaches used for the action segmentation and recognition task can be classified as either generative or discriminative models. The approaches presented in [19, 26] are generative models, while those presented in [33, 63] are discriminative models. Generative and discriminative models have complementary strengths. Generative models can easily deal with variable length sequences and missing data, while also being easier to design and implement [65, 87]. In contrast, discriminative models often achieve superior classification and generalisation performance [65, 87]. An ideal recognition system would hence combine these two separate but complementary approaches.

The Fisher vector (FV) approach [65, 36, 135] allows for the fusion of both generative and discriminative models. In contrast to the popular BoVW approach [169] which describes images by histograms of visual words, the FV approach describes images by deviations from a probabilistic visual vocabulary model. The resulting vectors can then be used by an SVM for final classification. Recently, FV has been successfully applied to the single-action recognition problem [112, 168].

A reliable low-level feature descriptor is a crucial stage for the success of an action recognition system. One popular descriptor for action recognition is Spatio-Temporal Interest Points (STIPs) [86]. However, STIP based descriptors have several drawbacks [22, 53]: they are computationally expensive, unstable, imprecise and can result in unnecessarily sparse detections. See Fig. 6.2 for a demonstration of STIP based detection. Other feature extraction techniques used for action recognition include gradients and optical flow [10, 53]. Each pixel in the gradient image helps extract relevant information, eg. edges (see Fig. 6.2). Since the task of action recognition is based on a sequence of frames, optical flow provides an efficient way of capturing the local dynamics [53].

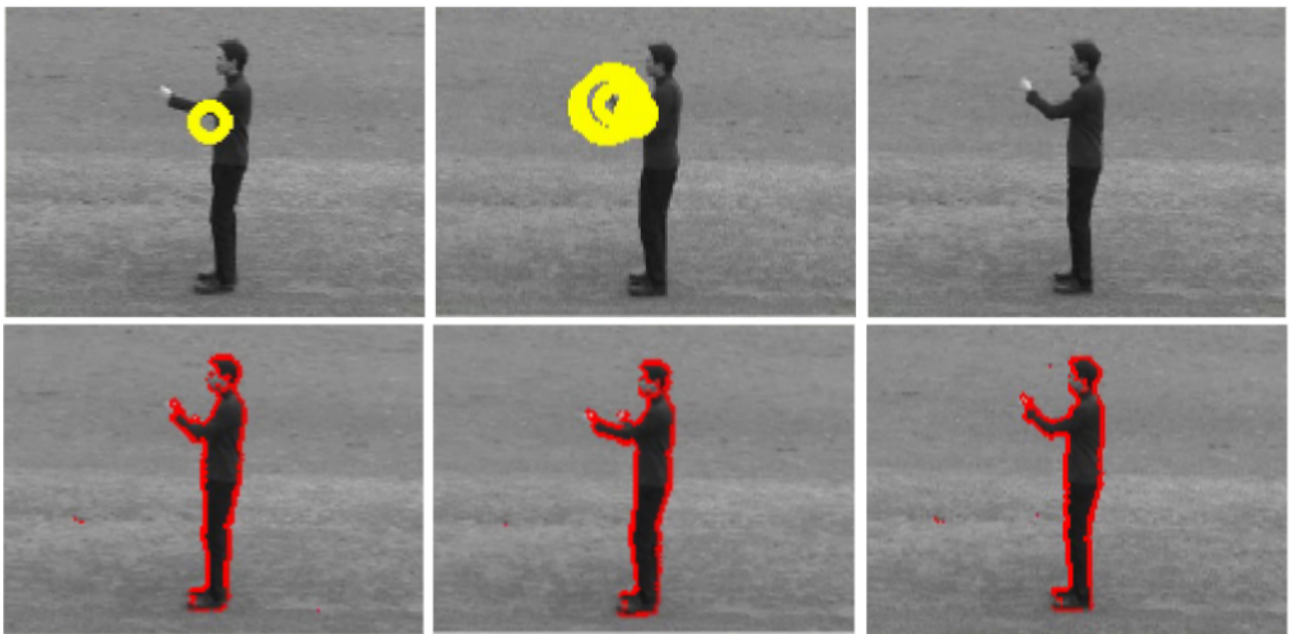


Figure 6.2: Top row: feature extraction based on Spatio-Temporal Interest Points (STIPs) is often unstable, imprecise and overly sparse. Bottom row: interest pixels (marked in red) obtained using magnitude of gradient.

Dense trajectories (DTs) have become popular in recent years. DT captures the local motion information of the video trajectories [167]. DT extracts the following descriptors: point coordinates (shape), histograms of oriented gradients (appearance), histograms of optical flow (motion), and motion boundary histograms (differential optical flow). The extracted descriptors are then aligned with the trajectories. Improved dense trajectories (IDT) were later proposed to remove trajectories caused by camera motion [168]. However, it has been reported that dense sampling is unable to distinguish between objects of interest and background [104]. For this reason, IDT agglomerates lots of unnecessary information, that can affect the learning process. This can directly affect large datasets in term of video duration, resolution and also number of classes. Moreover, for the aforementioned reason, IDT is expensive in terms of computation and computer data storage [179].

To the best of our knowledge, the combination of probabilistic integration with Fisher vectors or GMM is novel for the action segmentation and recognition problem. In contrast to [19, 63, 33], the proposed system requires fewer parameters to be optimised. We also avoid the need for a custom dynamic programming definition as in [63, 33]. Unlike our proposed GMM based approach the proposed method based on FV requires only one GMM for all actions,making it considerably more efficient. Moreover, proposed system based on FV combines the benefits of generative and discriminative models.

The chapter is continued as follows. We describe the proposed methods in Sections 6.2 and 6.3. Experiments and evaluation against a previous action segmentation and recognition methods is presented in Sections 6.5 and 6.6. Datasets are described in Section 6.4. The main findings and potential areas for future work are given in Section 6.7.

6.2 Proposed Method using Probabilistic Integration with Fisher Vectors

The proposed system using probabilistic integration with Fisher Vectors (**PI-FV**) have a hierarchical nature, stemming from progressive reduction and transformation of information, starting at the pixel level. The system is comprised of four main components:

(i) Division of a given video into overlapping multi-frame temporal windows, followed by extracting interesting low-level spatio-temporal features from each frame in each window.

(ii) Pooling of the interesting features from each temporal window to generate a sequence of Fisher vectors.

(iii) Conversion of each Fisher vector into a vector of class probabilities with the aid of a multi-class SVM.

(iv) Integration of the class probabilities at the frame level, leading to the final classification decision (action label) for each frame.

Each of the components is explained in more detail in the following subsections. The diagram of blocks of the proposed approached based on FV is shown in Fig 6.3.

6.2.1 Overlapping and Selective Feature Extraction

A video $\mathcal{V} = (\boldsymbol{I}_t)_{t=1}^{T}$ is an ordered set of T frames. We divide $\mathcal{V}$ into a set of overlapping temporal windows $(\mathcal{W}_s)_{s=1}^{S}$, with each window having a length of L frames. To achieve overlapping, the start of each window is one frame after the start of the preceding window. Each temporal window is hence defined as a set of frame identifiers:

$$\mathcal{W}_s = (t_{\text{start}},\ \ldots,\ t_{\text{start}-1+L}) \tag{6.1}$$

Each frame $\boldsymbol{I}_t \in \mathbb{R}^{r \times c}$ can be represented by a set of feature vectors $F_t = \{\boldsymbol{f}_p\}_{p=1}^{N_t}$ (with $N_t < r \cdot c$) corresponding to interesting pixels. We first extract the $d = 14$ dimensional feature vector $\boldsymbol{f}$ for each pixel in a given frame t as explained in Section 3.1.1.

Typically only a subset of the pixels in a frame correspond to the object of interest. As such, we are only interested in pixels with a gradient magnitude greater than a threshold β [54]. We discard feature vectors from locations with a small magnitude. In other words, only feature vectors corresponding to interesting pixels are kept. This typically results in a variable number of feature vectors per frame. See the bottom part in Fig. 6.2 for an example of the retained pixels.

6.2.2 Representing Windows as Fisher Vectors

Given a set of feature vectors, the Fisher Vector approach encodes the deviations from a probabilistic visual dictionary, which is typically a diagonal GMM. The parameters of a GMM with K components can be expressed as

$$\lambda = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\sigma}_k\}_{k=1}^{K}, \tag{6.2}$$

where, w_k is the weight, $\boldsymbol{\mu}_k$ is the mean vector, and $\boldsymbol{\sigma}_k$ is the diagonal covariance matrix for the k-th Gaussian.

The parameters are learnt using the Expectation Maximisation algorithm [18] on training data. For each temporal window $\mathcal{W}_s$, the feature vectors are pooled into set X^s containing $N = \sum_{t \in \mathcal{W}_s} N_t$ vectors. The deviations from the GMM are then accumulated using $\mathcal{G}_{\mu_k}^{X}$ and $\mathcal{G}_{\sigma_k}^{X}$ as explained in Section 3.3. The Fisher vector for window $\mathcal{W}_s$ is represented as the concatenation of $\mathcal{G}_{\mu_k}^{X}$ and $\mathcal{G}_{\sigma_k}^{X}$ (for $k = 1, \ldots, K$) into vector $\boldsymbol{\Phi}_s$. As $\mathcal{G}_{\mu_k}^{X}$ and $\mathcal{G}_{\sigma_k}^{X}$ are d-dimensional, $\boldsymbol{\Phi}_s$ has the dimensionality of $2dK$. Note that we have omitted the deviations for the weights as they add little information [135].

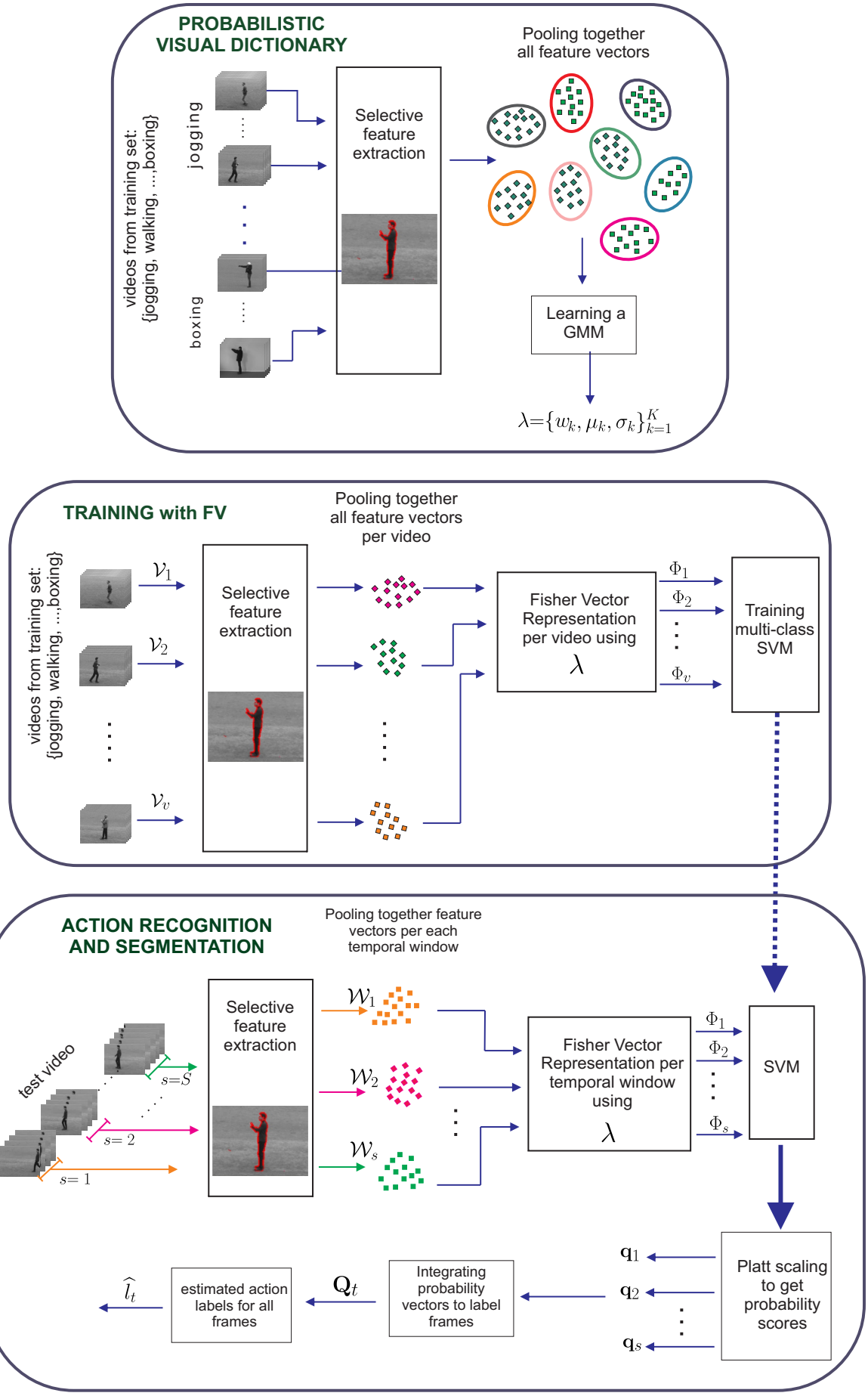


Figure 6.3: Proposed Method for Action Recognition and Segmentation using **PI-FV**.

6.2.3 Generation of Probability Vectors

For each Fisher vector we generate a vector of probabilities, with one probability per action class. First, a multi-class SVM [35] is used to predict class labels, outputting a set of raw scores. The scores are then transformed into a probability distribution over classes by applying Platt scaling [123]. The final probability vector derived from Fisher vector $\mathbf{\Phi}_s$ is expressed as:

$$\mathbf{q}_s = [\ P(l = 1|\mathbf{\Phi}_s),\ \cdots,\ P(l = A|\mathbf{\Phi}_s)\ ] \tag{6.3}$$

where l indicates an action class label and A is the number of action classes. The parameters for the multi-class SVM are learnt using Fisher vectors obtained from pre-segmented actions in training data.

6.2.4 Integrating Probability Vectors to Label Frames

As the temporal windows are overlapping, each frame is present in several temporal windows. We exploit the overlapping to integrate the class probabilities at the frame level. The total contribution of the probability vectors to each frame t is calculated by:

$$\mathbf{Q}_t = \sum\nolimits_{s=1}^{S} \mathbf{1}_{\mathcal{W}_s}(t) \cdot \mathbf{q}_s \tag{6.4}$$

where $\mathbf{1}_{\mathcal{W}_s}(t)$ is an indicator function, resulting in 1 if $t \in \mathcal{W}_s$, and 0 otherwise. The estimated action label for frame t is then calculated as: $\widehat{l}_t = \underset{l=1,\ldots,A}{\arg\max}\ \mathbf{Q}_t^{[l]}$, where $\mathbf{Q}_t^{[l]}$ indicates the l-th element of $\mathbf{Q}_t$.

6.3 Proposed Method using Probabilistic Integration with GMM

The proposed system using using probabilistic integration with GMM (**PI-GMM**) shares some similarities to the proposed method using FV. The diagram of blocks for the proposed PI-GMM approach is shown in Fig 6.4.

For PI-GMM, we learnt one model per action (See Section 3.2 for a detailed explanation of GMM). This results in a set of GMM models that we will express as $\{\lambda_a\}_{a=1}^{A}$, where A is the total number of actions. We also divide $\mathcal{V}$ into a set of overlapping temporal windows $(\mathcal{W}_s)_{s=1}^{S}$, with each window having a length of L frames. Features are extracted in the same fashion as in the proposed method using FV. For each temporal window $\mathcal{W}_s$, the feature vectors are pooled into set X^s containing $N = \sum_{t \in \mathcal{W}_s} N_t$ vectors. The feature vectors X are assumed independent, so the average log-likelihood of a model λ_a is computed as:

$$\log p(X^s|\lambda_a) = \sum\nolimits_{i=1}^{N} \log p(\boldsymbol{f}_i|\lambda_a), \tag{6.5}$$

where $\log p(X^s|\lambda_a)$ is calculated as in Eq. (3.5). Generally, the average log-likelihood is used by dividing $\log p(X^s|\lambda_a)$ by N, in order to normalise for varying lengths [132].

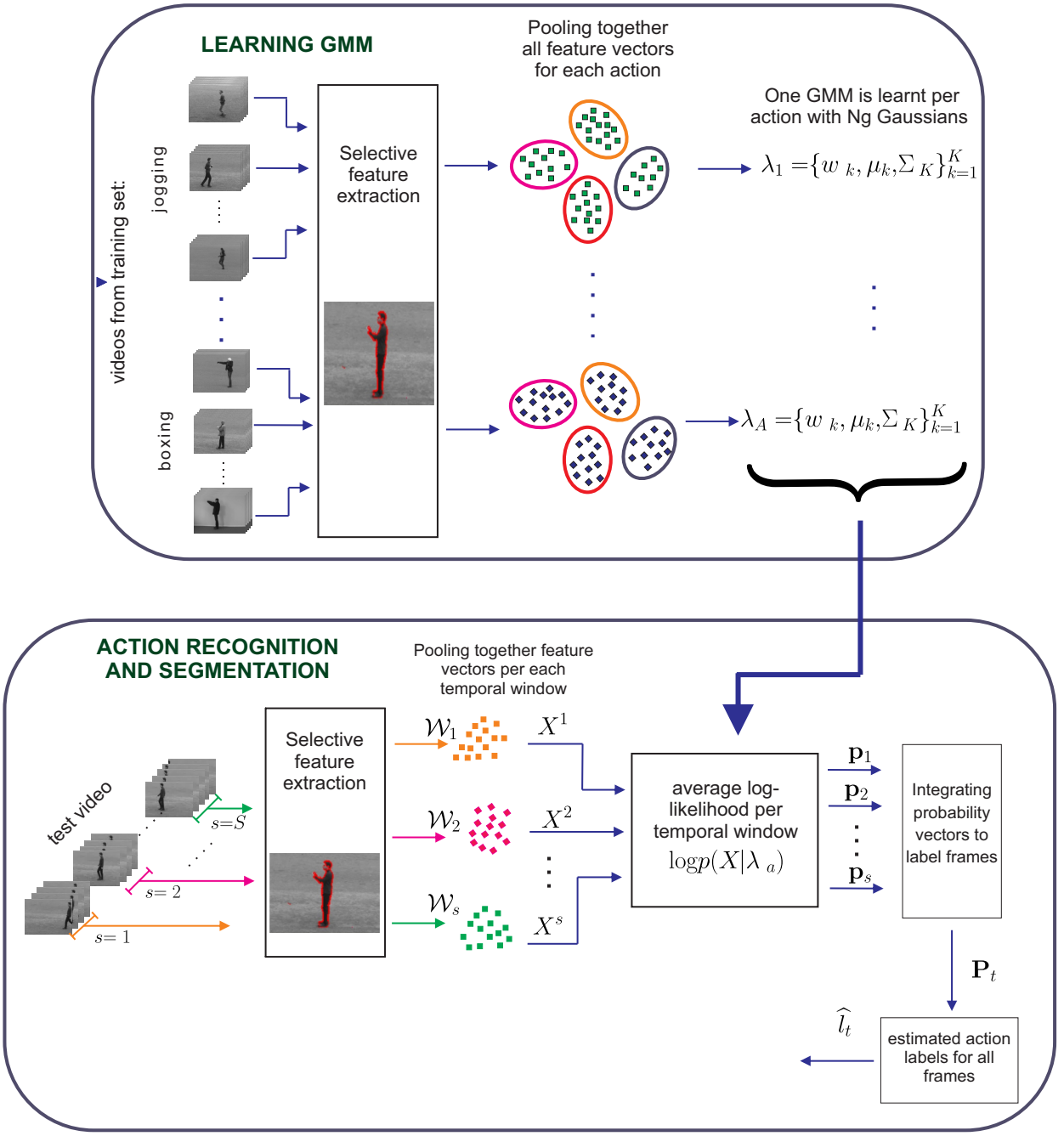


Figure 6.4: Proposed Method for Action Recognition and Segmentation using **PI-GMM**.

In each temporal window $\mathcal{W}s$, we compute the average log-likelihood for each model λ_a by using Eq. (6.5). The final probability vector derived from GMM $\mathbf{p}_s$ is obtained:

$$\mathbf{p}_s = [\ \log p(X^s|\lambda_1),\ \cdots,\ \log p(X^s|\lambda_A)\ ] \tag{6.6}$$

This vector $\mathbf{p}_s$ is the GMM equivalent to the final probability vector derived from Fisher vector $\mathbf{q}_s$ (Eq. 6.3). Similarly, the total contribution of the probability vectors to each frame t is calculated by:

Figure 6.5: Example of a multi-action sequence in the stitched version of the KTH dataset (s-KTH): boxing, jogging, hand clapping, running, hand waving and walking.

$$\mathbf{P}_t = \sum\nolimits_{s=1}^{S} \mathbf{1}_{\mathcal{W}_s}(t) \cdot \mathbf{p}_s \tag{6.7}$$

The estimated action label for frame t is then calculated as:

$$\widehat{l}_t = \underset{l=1,\ldots,A}{\arg\max}\, \mathbf{P}_t^{[l]} \tag{6.8}$$

where $\mathbf{P}_t^{[l]}$ indicates the l-th element of $\mathbf{P}_t$.

6.4 Datasets

We evaluated our proposed methods for joint action segmentation and recognition on two datasets: **(i)** a stitched version of the KTH dataset [140], and **(ii)** the challenging Carnegie Mellon University Multi-Modal Activity Dataset (CMU-MMAC) [40]. The results are reported in terms of frame-level accuracy as the ratio between the number of matched frames over the total number of frames.

The **s-KTH** (stitched KTH) dataset is obtained by simply concatenating existing single-action instances into sequences [19]. The KTH dataset contains 25 subjects performing 6 types of human actions and 4 scenarios (See Section 4 for more details). Each original video of the KTH dataset [140] contains an individual performing the same action. This action is performed four times and each subdivision or action-instance (in terms of start-frame and end-frame) is provided as part of the dataset. The action-instances (each video contains four instances of the action) were picked randomly, alternating between the two groups of {boxing, hand-waving, hand-clapping} and {walking, jogging, running} to accentuate action boundaries. See Fig. 6.5 for an example. The dataset was divided into two sets as in [19]: one for training and one for testing. In total, 64 and 36 multi-action videos were used for training and testing, respectively. We used 3-fold cross-validation, in contrast to [19] where one one validation is carried out.

The **CMU-MMAC** dataset is considerably more challenging as it contains realistic multi-action videos [40]. A kitchen was built to record subjects preparing and cooking food according to five recipes. See Section 4.4 for additional details. This dataset has occlusions, a cluttered background, and many distractors such as objects being deliberately moved. For our experiments we have used the same subset and camera view as per [19], which contains 12 subjects making brownies. The subjects were asked to make brownies in a natural way (no instructions were given). Each subject making the brownie is partially seen, as shown in Fig. 6.6. The videos have a high resolution and are

Figure 6.6: Example of a challenging multi-action sequence in the CMU-MMAC kitchen dataset: crack, read, stir, and switch-on.

longer than in s-KTH. The image size is 1024×768 pixels, and temporal resolution is 30 frames per second. The average duration of a video is approximately 15,000 frames and the average length of an action instance is approximately 230 frames (7.7s), with a minimum length of 3 frames (0.1s) and a maximum length of 3,269 frames (108s) [19]. The dataset was annotated using 14 labels, including the actions *close*, *crack*, *open*, *pour*, *put*, *read*, *spray*, *stir*, *switch-on*, *take*, *twist-off*, *twist-on*, *walk*, and the remaining actions (eg. frames in between two distinct actions) were grouped under the label *none* [150]. We used 12-fold cross-validation, using one subject for testing on a rotating basis.

6.5 Experiments with PI-GMM

All videos from both datasets were converted into gray-scale. Additionally, the videos from the CMU-MMAC dataset were re-scaled to 128×96 to reduce computational requirements. Based on preliminary experiments on both datasets, we used $\beta = 40$, where β is the threshold used for selection of interesting low-level feature vectors. Although the interesting feature vectors are calculated in all frames, we only use the feature vectors extracted from every second frame in order to speed up processing.

For the proposed PI-GMM, we have tuned the parameters using only the s-KTH dataset. As we will show in Section 6.6.2, PI-GMM is computationally more expensive than PI-FV. For this reason, an initial set of experiments has been performed to find the optimal number of components K. Using a fixed number of components $K = \{16, 64, 256, 1024\}$. We have evaluate the performance on one fold of the s-KTH dataset. For our experiments with PI-GMM, we used diagonal covariance matrices. GMM parameters were estimated using descriptors obtained from training videos using the iterative Expectation-Maximisation (EM) algorithm [18]. The duration of each segment L was set to 25 frames (1 second), which is the minimum length of an action-instance in the KTH dataset [14]. The results are reported in frame-level accuracy ($\%$) in Table 6.1.

We found that using $K = 1024$ provided optimal performance (77.0%). This better performance attained with 1024 components is explained by the fact that GMMs with large number of components are known to have the ability to model any given probability distribution function [22]. We kept the number of Gaussians constant for the remainder of our experiments with PI-GMM.

Table 6.1: Comparison of one run testing for several number of Gaussians (K) for PI-GMM.

K	**Accuracy (%)**
16	71.1
64	73.2
256	75.3
1024	77.0

6.6 Experiments with PI-FV

Parameters for the probabilistic visual vocabulary using GMM were learned using a large set of descriptors obtained from training videos using the iterative Expectation-Maximisation algorithm [18]. Specifically, we randomly sampled 100,000 feature vectors for each action and then pooled all the resultant feature vectors from all actions for training. Experiments were performed with three separate visual vocabularies with varying number of components: $K = \{64, 128, 256\}$. We have not evaluated larger values of K due to increased computational complexity and hence the exorbitant amount of time required to process the large CMU-MMAC dataset, which contains on average 15,000 frames per video. To learn the parameters of the multi-class SVM, we used video segments containing single actions. For s-KTH this process is straightforward as the videos have been previously segmented. The CMU-MMAC dataset contains continuous multi-actions. For this reason, to train our system we obtain one Fisher vector per action in each video, using the low-level feature vectors belonging to that specific action.

6.6.1 Effect of Window Length and Dictionary Size

On account of FV representation is an evolution of the BoVW (as previously explained), we have evaluated not only the performance of PI-FV but also the variant probabilistic integration with BoVW histograms (PI-BoVW), where the Fisher vector representation is replaced with BoVW representation. We start our experiments by studying the influence of the segment length L, expressed in terms of seconds. The results are reported in Figs. 6.7 and 6.8, in terms of average accuracy over the folds.

Using the PI-FV variant (Fig. 6.7), we found that using $L = 1s$ and $K = 256$ leads to the best performance on the s-KTH dataset. For the CMU-MMAC dataset, the best performance is obtained with $L = 2.5s$ and $K = 64$. Note that using larger values of K (ie., 128 and 256) leads to worse performance. We attribute this to the large variability of appearance in the dataset, where the training data may not be a good representative of the test data. Consequently, using a large value of K may lead to overfitting to the training data.

The optimal segment length for each dataset is different. We attribute this to the s-KTH dataset containing short videos whose duration is between $1s$ and $7s$, while CMU-MMAC has a large range of action durations between $0.1s$ and $108s$. While the optimal values of L and K differ across the

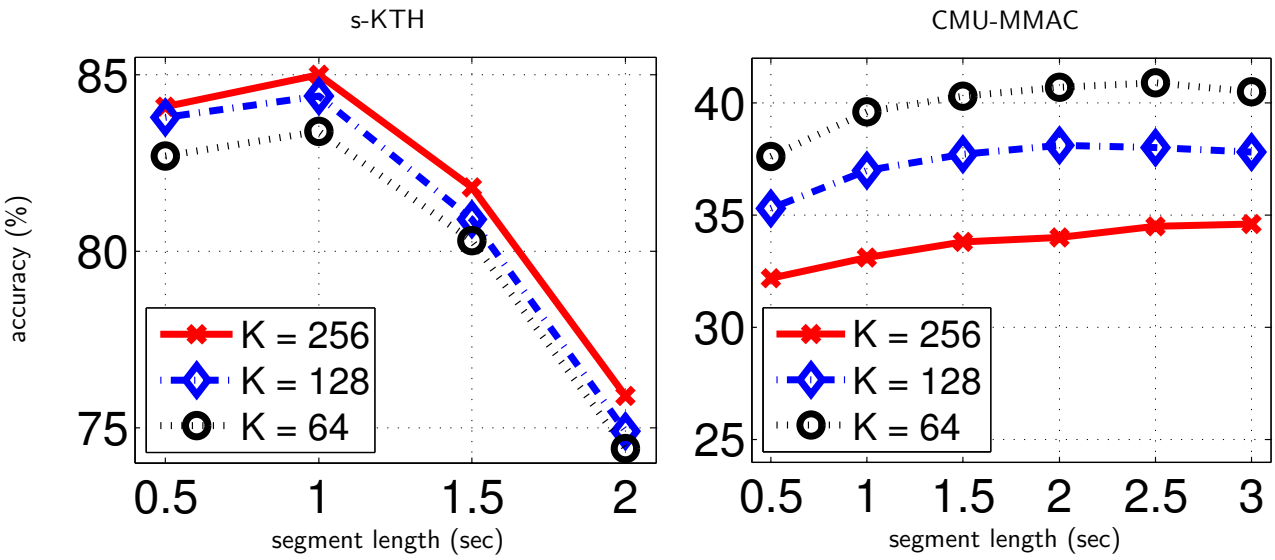


Figure 6.7: Performance of the proposed **PI-FV** approach for varying the segment length on the s-KTH and CMU-MMAC datasets, in terms of average frame-level accuracy over the folds.

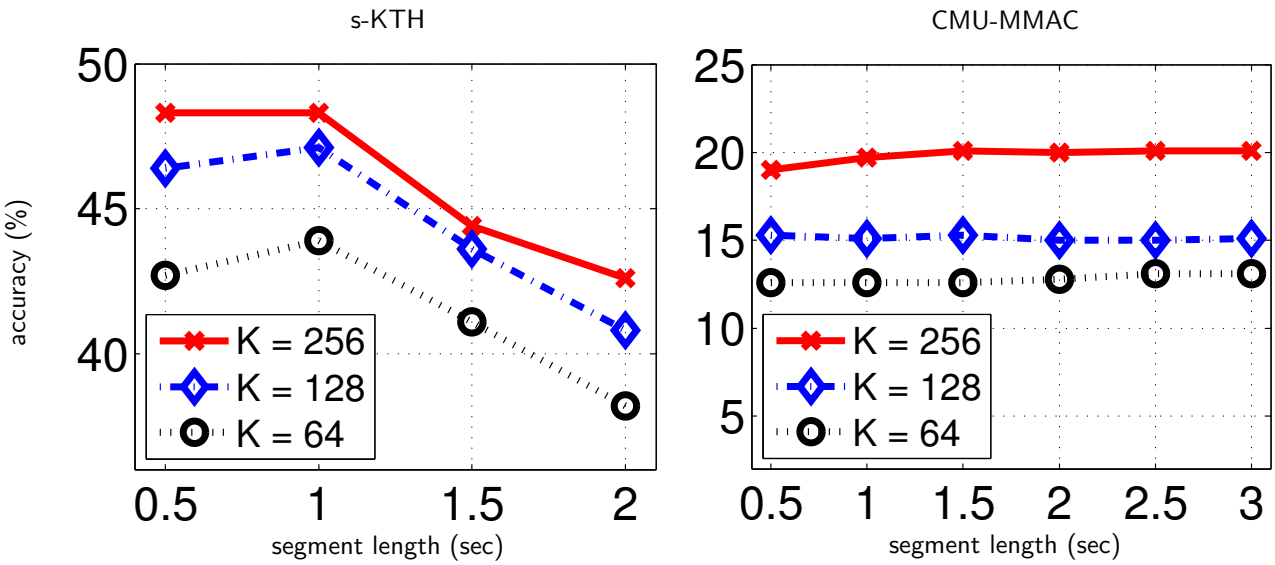


Figure 6.8: As per Fig. 6.7, but showing the performance of the **PI-BoVW** variant (where the Fisher vector representation is replaced with BoVW representation).

datasets, the results also show that relatively good overall performance across both datasets can be obtained with $L = 1s$ and $K = 64$.

The results for the PI-BoVW variant are shown in Fig. 6.8. The best performance for the PI-BoVW variant on the s-KTH dataset is obtained using $L = 1s$ and $K = 256$, while on the CMU-MMAC dataset it is obtained with $L = 2.5s$ and $K = 256$. These are the same values of L and K as for the PI-FV variant. However, the performance of the PI-BoVW variant is consistently worse than the PI-FV variant on both datasets. This can be attributed to the better representation power of FV. Note that the visual dictionary size K for BoVW is usually higher in order to achieve performance similar to FV. However, due to the large size of the CMU-MMAC dataset, and for direct comparison purposes, we have used the same range of K values throughout the experiments.

Figs. 6.9 and 6.10 show qualitative examples of segmentation on the s-KTH and CMU-MMAC datasets, respectively. It can be seen that the PI-FV variant obtains qualitatively more accurate segmentation. Figs. 6.11 and 6.12 show the confusion matrices for the PI-FV and PI-BoVW variants on the CMU-MMAC dataset. The confusion matrices show that in 50% of the cases (actions), the PI-BoVW variant is unable to recognise the correct action. Furthermore, the PI-FV variant on average obtains better action segmentation than PI-BoVW.

For five actions (*crack*, *open*, *read*, *spray*, *twist-on*), PI-FV has accuracies of $\leq 0.5\%$. Action *crack* implies crack and pour eggs into a bowl, but it's annotated only as *crack*, leading to confusion between *crack* and *pour*. We suspect that actions *read* and *spray* are poorly modelled due to lack of training data; they are performed by a reduced number of subjects. Action *twist-on* is confused with *twist-off* which are essentially the same action.

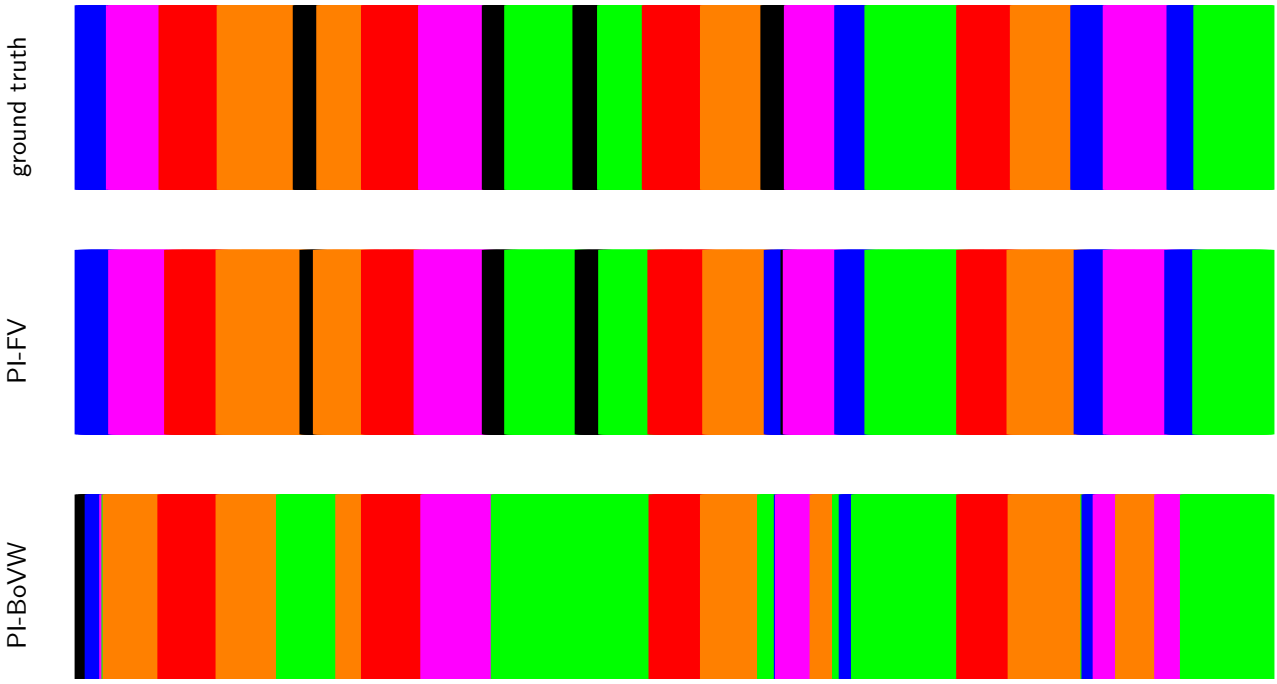


Figure 6.9: Qualitative example of segmentation using PI-FV and PI-BoVW versus ground truth on the s-KTH dataset. Each colour represents a unique action.

6.6.2 Comparison with PI-GMM and HMM-MIO

We have compared the performance of the PI-FV and PI-BoVW variants against the proposed PI-GMM and HMM-MIO [19] previously used for action recognition and segmentation. For PI-GMM, we have used on both datasets the optimal parameters found in Section 6.5. The comparative results are shown in Table 6.2.

The proposed PI-FV method obtains the highest accuracy of 85.0% and 40.9% for the s-KTH and CMU-MMAC datasets, respectively. In addition to higher accuracy, the proposed method has other advantages over previous techniques. There is just one global GMM (representing the visual

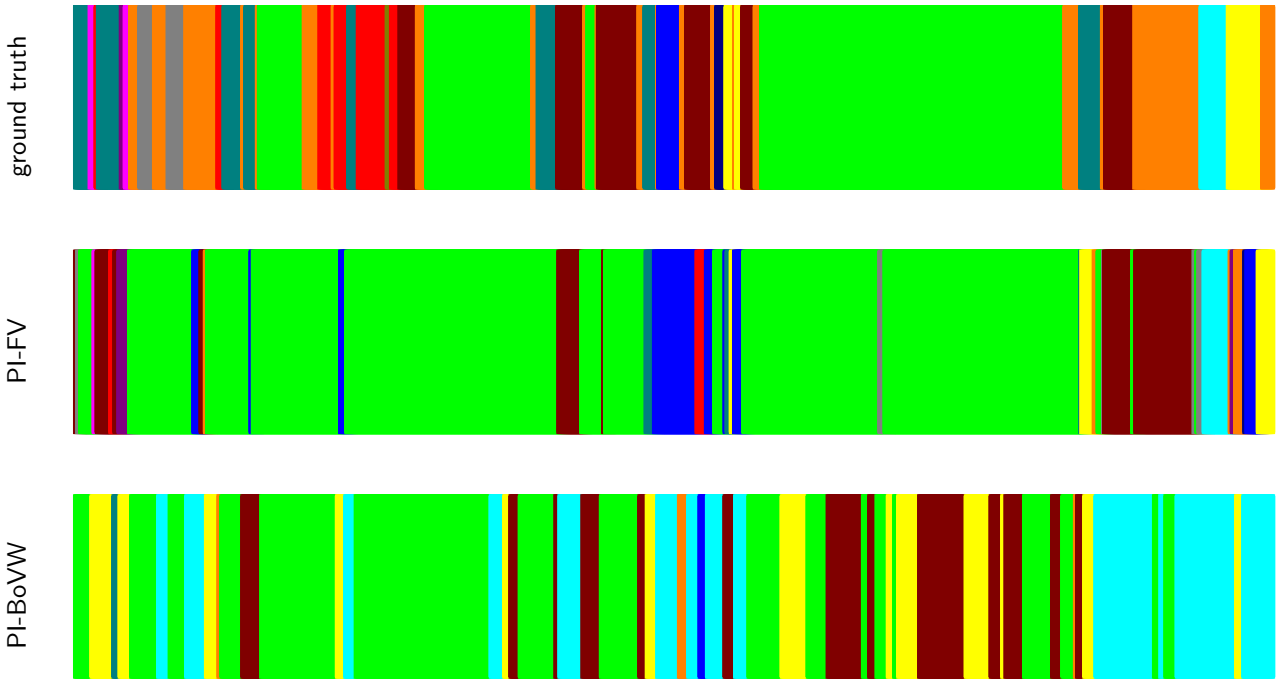


Figure 6.10: As per Fig. 6.9, but on the difficult CMU-MMAC dataset.

vocabulary). This is in contrast to proposed PI-GMM which uses one GMM (with a large number of components) for each action, leading to high computational complexity. The HMM-MIO method in [19] requires the search for many optimal parameters (as described in Section 6.1), whereas the proposed method has just two parameters (L and K).

Table 6.2: Comparison of the proposed methods (PI-FV and PI-BoVW) against several recent approaches on the stitched version of the KTH dataset (s-KTH) and the challenging CMU-MMAC dataset.

Method	**s-KTH**	**CMU-MMAC**
HMM-MIO [19]	71.2%	38.4%
PI-GMM	78.3%	33.7%
PI-FV	**85.0%**	**40.9%**
PI-BoW	48.0%	20.1%

Lastly, we provide an analysis of the computational cost (in terms of wall-clock time) of our system and the stochastic modelling approach. The wall-clock time is measured under optimal configuration for each system, using a Linux machine with an Intel Core processor running at 2.83 GHz. On the s-KTH dataset, the PI-GMM system takes on average 228.4 minutes to segment and recognise a multi-action video. In comparison, the PI-FV takes 5.6 minutes, which is approximately 40 times faster.

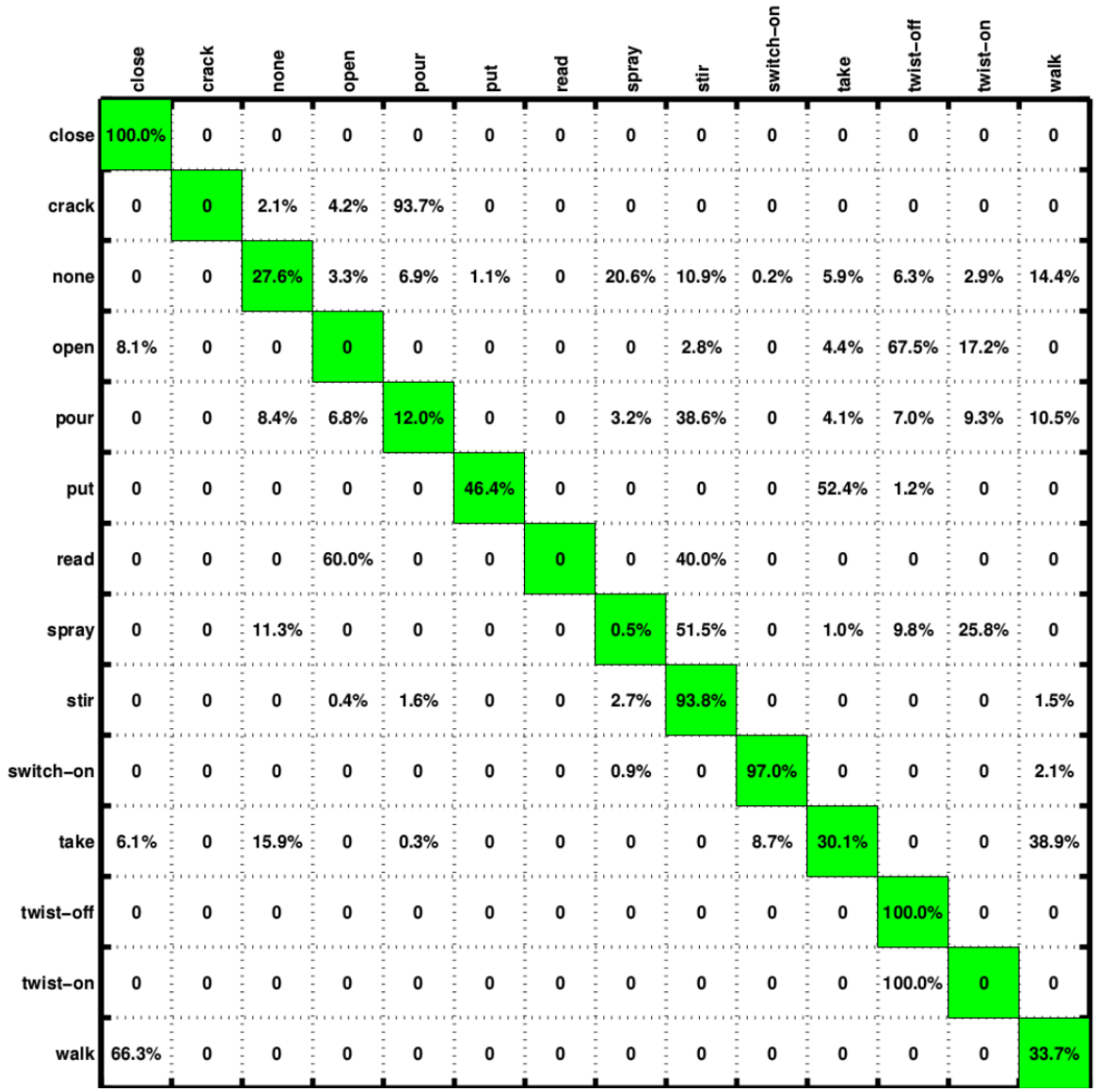


Figure 6.11: Confusion matrix for the PI-FV variant on the CMU-MMAC dataset.

6.7 Conclusions

In this chapter we have proposed two hierarchical approaches that perform joint action segmentation and classification in videos: PI-FV and PI-GMM. Videos are processed through overlapping temporal windows. Each frame in a temporal window is represented using selective low-level spatio-temporal features which efficiently capture relevant local dynamics and do not suffer from the instability and imprecision exhibited by STIP descriptors [86]. For the PI-FV, features from each window are represented as a Fisher vector, which captures the first and second order statistics. Rather than directly classifying each Fisher vector, it is converted into a vector of class probabilities. For PI-GMM, the vector of class probabilities is obtained using the average log-likelihood over each temporal window. The final classification decision for each frame (action label) is then obtained by integrating the class probabilities at the frame level, which exploits the overlapping of the temporal windows. The proposed approach has a lower number of free parameters than previous methods which use dynamic

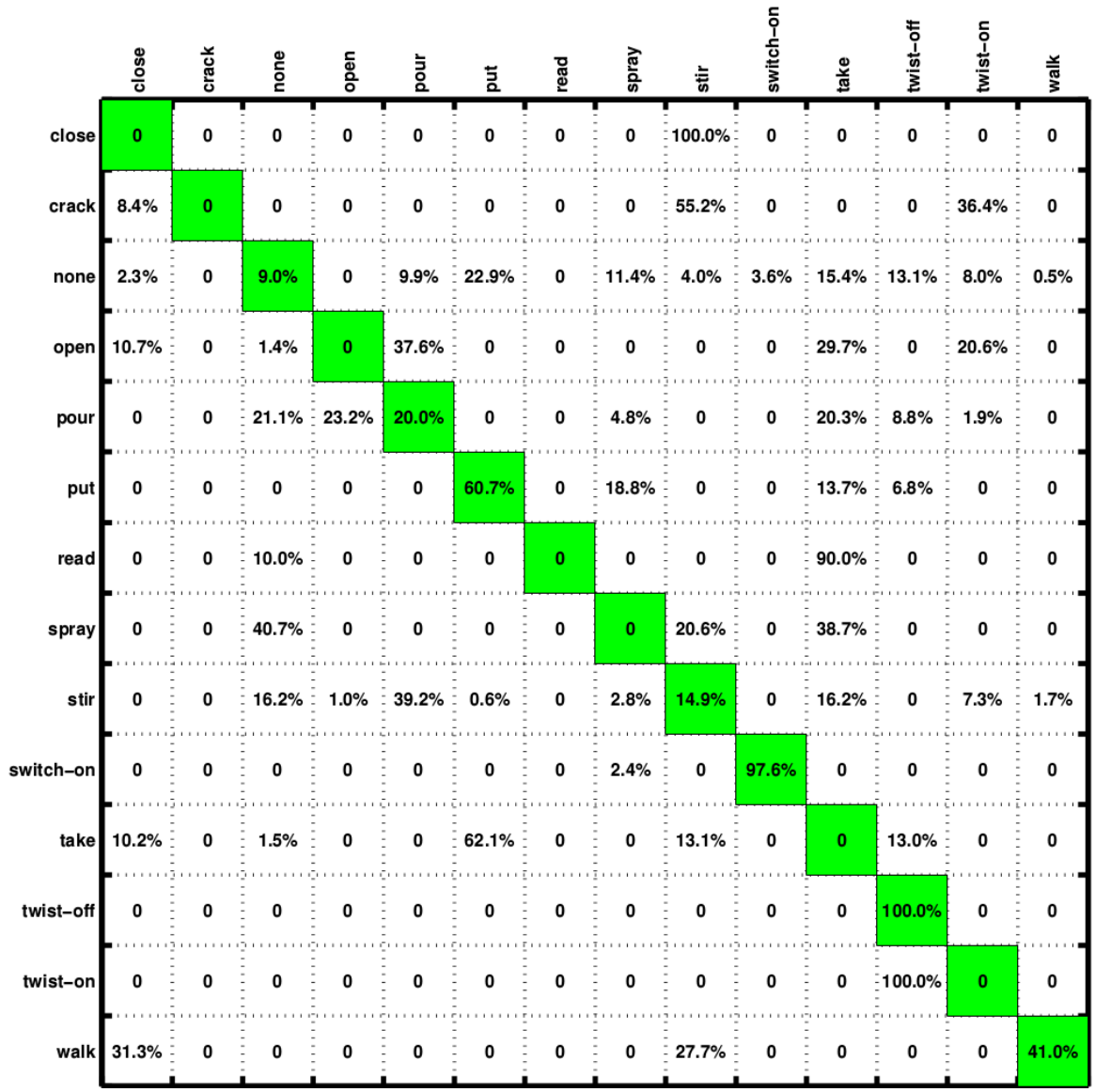


Figure 6.12: As per Fig. 6.11, but using the PI-BoVW variant.

programming or HMMs [19]. We have found that PI-FV it is also considerably less computationally demanding compared to modelling each action directly with PI-GMM.

Experiments were done on two datasets: s-KTH (a stitched version of the KTH dataset to simulate multi-actions), and the more challenging CMU-MMAC dataset (containing realistic multi-action videos of food preparation). On s-KTH, the proposed PI-FV achieves an accuracy of 85.0%, considerably outperforming proposed PI-GMM and HMM-based approach which obtained 78.3% and 71.2%, respectively. On CMU-MMAC, the proposed approach achieves an accuracy of 40.9%, outperforming the PI-GMM and HMM approaches which obtained 33.7% and 38.4%, respectively. Furthermore, the proposed system PI-FV is on average 40 times faster than the also proposed PI-GMM approach.

Chapter 7

Towards Miss Universe Automatic Prediction via Catwalk Analysis

If women want to ensure themselves a meaningful place in the future, they need to be among those determining how the technology will be used. They need to be among those deciding whether it will be the great leveller or simply serve to worsen social divisions.

Anita Borg

Can we predict the winner of Miss Universe after watching how they stride down the catwalk during the evening gown competition? Fashion gurus say they can!

In this chapter[1], we study this question from the perspective of computer vision. In particular, we want to understand whether existing computer vision approaches can be used to automatically extract the qualities exhibited by the Miss Universe winners during their catwalk. This study can pave the way towards new vision based applications for the fashion industry. To this end, we propose a novel video dataset, called the Miss Universe dataset, comprising 10 years of the evening gown competition selected between 1996-2010. We further propose two ranking-related problems: (1) the Miss Universe Listwise Ranking and (2) the Miss Universe Pairwise Ranking problems. In addition, we also develop an approach that simultaneously addresses the two proposed problems. To describe the videos we employ the recently proposed Stacked Fisher Vectors in conjunction with robust local spatio-temporal features. From our evaluation we found that although the addressed problems are extremely challenging, the proposed system is able to rank the winner in the top 3 best predicted scores for 5 out of 10 Miss Universe competitions.

[1]The work presented in this chapter has been published in [28].

7.1 Introduction

Miss Universe is a worldwide pageant competition held every year since 1952 and is organised by *The Miss Universe Organization* [2]. Every year up to 89 candidates participate in the competition. Each delegate must first win their respective national pageants. Miss Universe is broadcast in more than 190 countries around the world and is watched by more than half a billion people annually [2, 3]. Although Miss Universe is one of the most publicised beauty pageants in the world, it is not the only existing pageant competition. A list of beauty pageants from around the world includes up to 22 events among international, continental and, regional pageants. Moreover, there are more than 260 national pageants. In the US alone, there are approximately 28 national pageants [1].

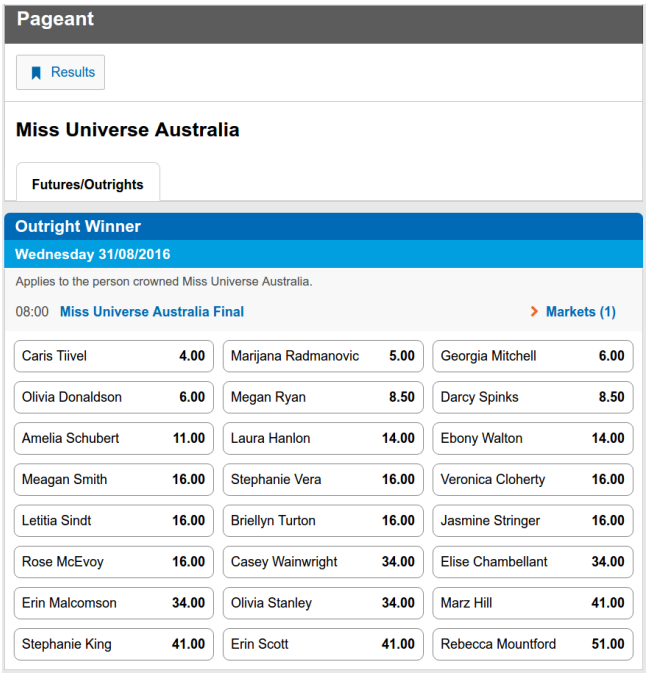
The format has slightly changed during the 64 year period. However, the most common competition format is as follows. All candidates are preliminary judged in three areas of competition: Interview, Swimsuits and Evening Gown. After that, the top 10 or 15 semi-finalists are short-listed during the coronation night. The semi-finalists compete again in swimsuits and evening gowns. The best 5 finalists are selected and go through an interview round. Finally, the runners-up and winner are announced. During the swimsuit and evening gown competition, the catwalk is judged by several aspects. Candidates must emanate poise, posture, grace, elegance, balance, confidence, energy, charisma, and sophistication. Additionally, during the swimsuit competition candidates are expected to have a well-proportioned body, good muscle tone, proper level of body fat and show fitness and body shape.

In our work, we study the possibility of capturing these qualities to predict the winner. This can pave the way of numerous vision-based applications for the fashion industry such as automatic training systems for amateur models who aspire to become professionals. Due to the complexity of this problem, we propose to initially study the evening gown competition. To this end, we collect a new dataset of videos recorded during the evening gown competition where the judges' scores and recordings are publicly available.

As mentioned, there are many potential commercial application for an automatic system able to analyse and predict the best catwalk in a beauty pageant. Automatically predicting the winner can be useful for specialised betting sites such as Odds Shark, Sports Bet, Bovada, and Bet Online. These betting sites allow the audience to bet for their favourite candidate in Miss Universe. Fig. 7.1 shows the Miss Universe Australia betting site in Sports Bet. In this online betting site bettors can place a wager on the outcome of Miss Universe Australia.

Catwalk analysis can be also a powerful tool for boutique talent agencies such as "Polished by Donna" that provides training for improving the catwalk[2] and offer their services to future beauty pageants candidates (See Fig. 7.2). For boutique talent agencies, an automatic catwalk analysis system can help to compare the improvement of each client again herself or again an experienced catwalker.

[2] http://www.polishedbydonna.com/#!catwalk--pageant-training/c1drx



Pageant

Results

Miss Universe Australia

Futures/Outrights

Outright Winner

Wednesday 31/08/2016

Applies to the person crowned Miss Universe Australia.

08:00 **Miss Universe Australia Final** Markets (1)

Caris Tiivel	4.00	Marijana Radmanovic	5.00	Georgia Mitchell	6.00
Olivia Donaldson	6.00	Megan Ryan	8.50	Darcy Spinks	8.50
Amelia Schubert	11.00	Laura Hanlon	14.00	Ebony Walton	14.00
Meagan Smith	16.00	Stephanie Vera	16.00	Veronica Cloherty	16.00
Letitia Sindt	16.00	Briellyn Turton	16.00	Jasmine Stringer	16.00
Rose McEvoy	16.00	Casey Wainwright	34.00	Elise Chambellant	34.00
Erin Malcomson	34.00	Olivia Stanley	34.00	Marz Hill	41.00
Stephanie King	41.00	Erin Scott	41.00	Rebecca Mountford	51.00

Figure 7.1: Miss Universe Australia betting site in `www.sportsbet.com.au`. Bettors can can place a wager on his/her favourite candidate. The number in front of each participant for Miss Universe Australia means the estimate returns.

Catwalk & Pageant Training

Donna & her team are highly experienced in the areas of runway, catwalk, pageantry, bridal work and fitness competitions.

Previous clients have competed in major beauty pageants including *Miss Universe*, *Miss World* and *Miss Earth*.

Donna has been involved in the choreography of countless numbers of fashion shows, both commercial and high fashion, locally, nationally and internationally.

Donna will work closely with you on your posture, balance, walk, posing and turns to get the best out of your catwalk.

If you have a major competition or runway performance coming up, we can even specifically choreograph routines for you giving you the confidence and competitive advantage.

We can choreograph individual, double or even group runway shows.

Please feel free to contact the agency for rates, training locations, potenital dates and for anymore information here.

"I came to Donna for a private tuition on catwalk as I needed to learn within two weeks. She was excited to help me out and was able to fit me in at short notice. I couldn't be happier with the lesson! I attended a model course a few years back. I learnt so much more this time with Donna, at a reasonable price as well. Donna was friendly and welcoming at the same time as professional with no judgement, made me feel more relaxed. I would definitely recommend the private tuition or any course that is offered by Polished by Donna!" - Debbie

Figure 7.2: Services offered by "Polished by Donna". Catwalk and Pageant Training.

Towards automatic prediction of Miss Universe and automatic catwalk analysis, we first collect a novel dataset named the Miss Universe (MU) dataset. The dataset comprises 10 years of Miss Universe selected from 1996 to 2010. The years included in this datasets are: 2010, 2007, 2003, 2002, 2001, 2000, 1999, 1998, 1997, and 1996. The years not included were due to the videos and/or the scores were not publicly available.

It comprises 105 videos and 18,343 frames depicting each candidate catwalk in the evening gown competition. The selected years of Miss Universe also include the official judges' scores. Fig. 7.3 shows two examples of best and worst judges' scores during the evening gown competition. We propose two sub-problems: (1) The Miss Universe Listwise Ranking (MULR) problem and (2) The Miss Universe Pairwise Ranking (MUPR) problem. While the former aims to predict the winner of the evening gown competition, the latter focuses on judging the catwalk between two participants. Note that the solution of the MUPR problem could be used for developing applications for boutique talent agencies.

In this work, we propose an approach which will address both problems simultaneously. More specifically, we found that it is possible to share the model trained from one problem with the other problem. We use our approach in conjunction with the video descriptors used for action analysis as explained in Section 3.1. In particular the video descriptors are extracted on a pixel-base and make use of gradients and optical flow. Gradients and optical flow have been shown to be effective for video representation. Then, the video descriptors are encoded using the Stacked Fisher Vectors (SFV) approach, which has recently shown successful performance for action analysis [117]. From our evaluations, we found that that our proposed problems are extremely challenging. However, further analysis suggests that both problems could still be potentially solved using a computer vision approach.

7.2 Problem Definition

During the evening gown competition, candidates are given an average score based on their catwalk. Different judges are selected each year to score each candidate. This score is used in conjunction with the swimming competition, to select the best 5 finalists, where finally the Miss Universe winner is announced. See Fig. 7.3 for examples of best and worst scores. Candidates with the best scores strut with attitude down the catwalk projecting confidence. Their arms are kept relaxed and swing naturally with the body. In general, they exhibit a flouncing walk and ooze elegance as they stalk the runway. Candidates with the worst scores make their arms too stiff, and look very robotic and awkward. They do not keep their body loose and sometimes droop their heads. It can be also seen that the candidate with the worst catwalk during Miss Universe 2010 finds herself struggling to walk with the ribbon dress that is too tight for her.

Our central problem is to predict the best catwalk during the evening gown competition. This can be considered as an instance of the ranking problem. The ranking problem has been explored in various domains such as collaborative filtering, documents retrieval, and sentiment analysis [23].

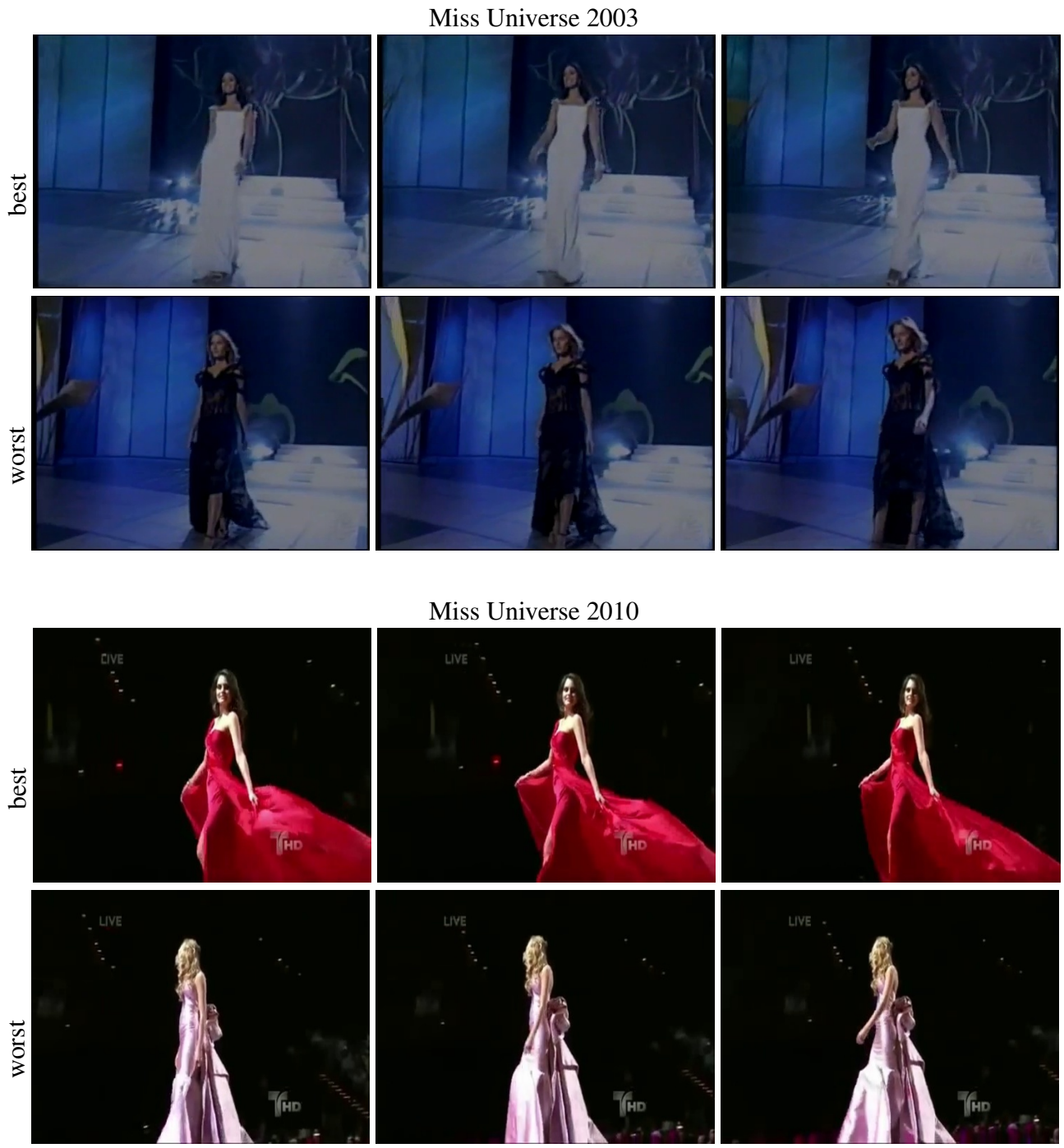


Figure 7.3: Examples of best and worst scores for Miss Universe versions 2003 and 2010.

In our work, we define two ranking sub-problems: (1) the Miss Universe Listwise Ranking (MULR) problem and (2) the Miss Universe Pairwise Ranking (MUPR) problem. While the MULR focuses on rank ordering of all Miss Universe participants in the same year, the MUPR considers pairwise comparisons of two participants in the same year. These two sub-problems have also been described in [32, 29] for general machine learning problems.

7.2.1 The Miss Universe Listwise Ranking Problem (MULR)

The MULR problem can be formalised as follows. Given a query $\mathcal{Q}_l = \{\boldsymbol{p}_j^{(q)}\}_{j=1}^{Nq}$, where $\boldsymbol{p}_j^{(q)}$ is the video of a participant for Miss Universe from year q and N_q is the total number of candidates for that specific year. Let $\mathcal{G}_l = \{\mathcal{S}_m\}_{m=1}^{M}$ be the gallery containing M sets of Miss Universe from M years, where $\mathcal{S}_m = \{\boldsymbol{p}_j^{(m)}\}_{j=1}^{N_m}$ is the set of N_m participant videos of Miss Universe from year m. Each set of participants $\mathcal{S}_m$ is associated with a set of judgments (scores) $\mathbf{y}^{(m)} = [y_1^{(m)}, \cdots, y_{N_m}^{(m)}]$. The judgment $y_j^{(m)}$ represents the average score of participant $\boldsymbol{p}_j^{(m)}$. The average score is calculated by averaging the scores given by all the judges during the evening gown competition. A set of video descriptors $\mathbf{v}_j^{(m)} = \Phi(\boldsymbol{p}_j^{(m)})$, where $\mathbf{v}_j^{(m)} \in \mathbb{R}^d$ are extracted from each participant video, $\boldsymbol{p}_j^{(m)}$.

Let $f_l : \mathbb{R}^d \mapsto \mathbb{R}^1$ be a scoring function that calculates a participant score based on its corresponding video descriptors. Given the query $\mathcal{Q}_l$, the function f_l can automatically score each participant in $\mathcal{Q}_l$. Let $\mathbf{y}^{(q)} = [y_1^{(m)}, \cdots, y_{N_q}^{(q)}]$ be the actual score from the judges for participants in the query $\mathcal{Q}_l$, and $\hat{\mathbf{y}}^{(q)} = [f(y)_1^{(m)}, \cdots, f(y)_{N_q}^{(q)}]$ be the estimated score of function f trained using the gallery set $\mathcal{G}_l$, the main task in MULR problem is to find the best f_l, where ideally the ranking of $\hat{\mathbf{y}}^{(q)}$ is the same as $\mathbf{y}^{(q)}$.

7.2.2 The Miss Universe Pairwise Ranking problem (MUPR)

For the MUPR problem, we first consider a gallery $\mathcal{G}_p$:

$$\mathcal{G}_p = \{(\boldsymbol{p}_l^{(m)}, \boldsymbol{p}_k^{(m)})\}_{m=1}^{M}, \boldsymbol{p}_l^{(m)}, \boldsymbol{p}_k^{(m)} \in \mathcal{S}_m, l \neq k \tag{7.1}$$

where each element in the gallery is a pair of participant videos from the same year of Miss Universe. Note that the gallery $\mathcal{G}_p$ considered in this problem is different from the gallery $\mathcal{G}_l$ considered in MULR problem. Each pair in the gallery has its corresponding label $y_{lk}^{(m)}$ which is defined via:

$$y_{lk}^{(m)} = \begin{cases} +1; & y_l^{(m)} > y_k^{(m)} \\ -1, & \text{otherwise} \end{cases}, \tag{7.2}$$

where $y_l^{(m)}$ and $y_k^{(m)}$ are the actual score from the judges. Let $(\boldsymbol{p}_l^{(q)}, \boldsymbol{p}_k^{(q)})$, y_{lk}^{q} be a query pair and its corresponding label, the main task for the MUPR problem is to find the best ranking function $f_p(\cdot) = \{-1, +1\}$ where ideally $y_{lk}^{(q)} = f_p(\boldsymbol{p}_l^{(q)}, \boldsymbol{p}_k^{(q)})$.

7.3 Proposed Approach

Here we present our approach to solve both MULR and MUPR problems simultaneously. We start explaining the video encoding using SFV. The, we reveal how we simultaneously address both MULR and MUPR problems using the same framework.

7.3.1 Video encoding via Stacked Fisher Vectors

The video descriptors used for Miss Universe are explained in Section 3.1.1. Those descriptors include the pixel coordinates and descriptors obtained from the image gradient and optical flow. Using the gradient magnitude, we select pixels that correspond to the object of interested. As video encoder a recent version of the traditional Fisher Vector is used. As explained in Section 3.3, the traditional FV consists in describing a pooled set of features by its deviation from a generative model. FV encodes the deviations from a probabilistic version of a visual dictionary, which is typically a Gaussian Mixture Model (GMM) with diagonal covariance matrices [120, 135]. The model is given by $\lambda = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\sigma}_k\}_{k=1}^{K}$, where, w_k is the weight, $\boldsymbol{\mu}_k$ is the mean vector, and $\boldsymbol{\sigma}_k$ is the diagonal covariance matrix for the k-th Gaussian.

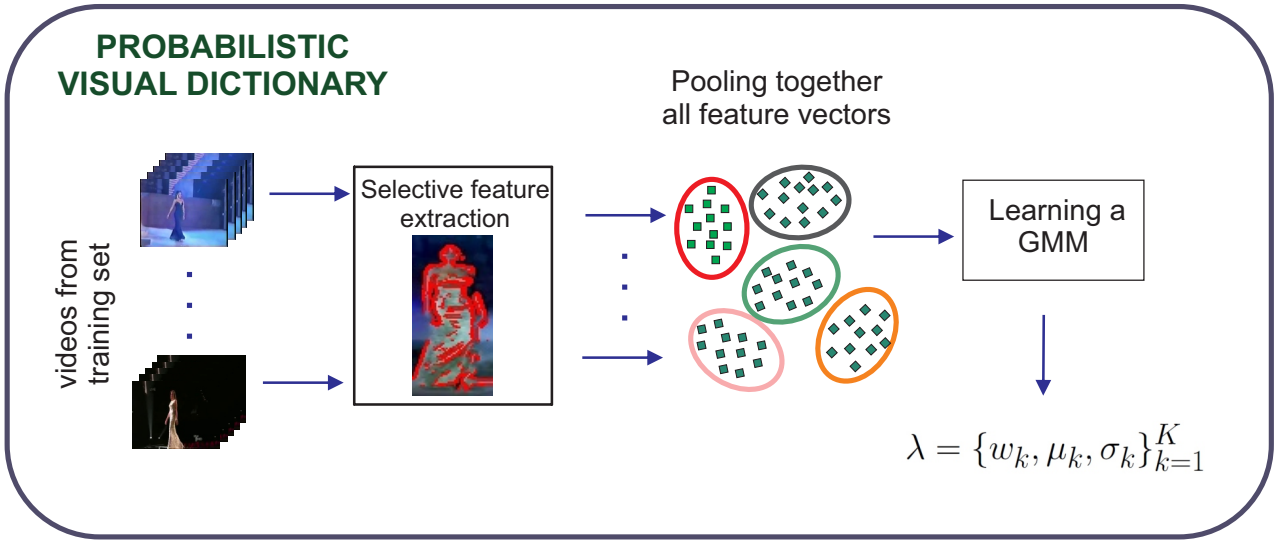


Figure 7.4: Probabilistic Visual Dictionary for first layer.

Stacked Fisher Vectors (SFV) is a multi-layer representation of the standard FV [117]. For SFV, we first performs traditional FV representation over densely sampled consecutive segments based on low level descriptors. Fig 7.4 shows how the probabilistic visual dictionary is learnt for the first layer . The extracted FVs have a high dimensionality and are fed the next layer. The second layer reduces the dimensionality of the obtained FVs, and then those reduced FVs are encoded again with FV representation. For this second layer another probabilistic visual dictionary is learnt. Finally, one SFV is obtained per each video. See Fig 7.5 for a description of the SFV process.

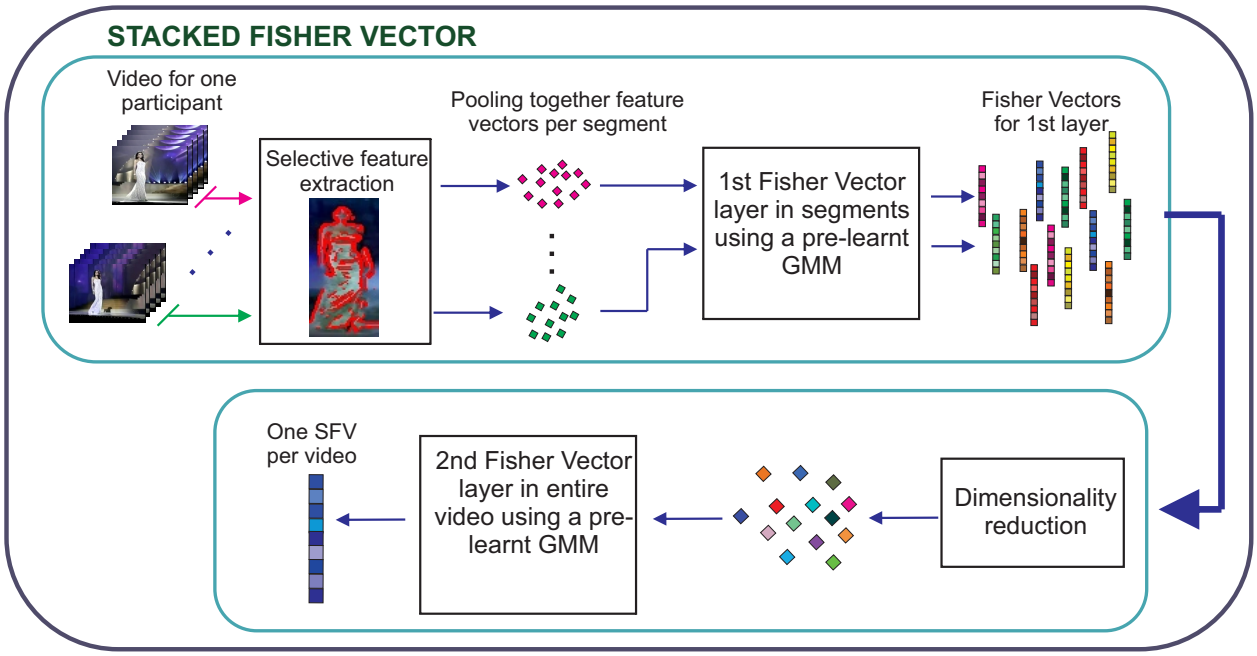


Figure 7.5: SFV is performed for each video. It comprises two layers. The first layer is the traditional FV over segments. Second layer encodes the obtained FV from the first layer. One SFV is obtained per video.

7.3.2 Classification

We address both MULR and MUPR problems using the same framework. Recall that the main objective of the MULR problem is to find the best $f_l(\cdot)$ wherein its scores can be used to rank the Miss Universe participants from the same year. We model such a function as a linear regression function defined as:

$$f_l(\mathbf{v}) = \mathbf{w}^\top \mathbf{v} + b, \tag{7.3}$$

where $\mathbf{w}$ and b are the parameters of the regression model and $\mathbf{v} \in \mathbb{R}^d$ is the extracted video descriptor after applying SFV. As it is not trivial to train the regression given the gallery $\mathcal{G}_l$ with its corresponding actual ranking, we solve this problem by addressing MUPR, which is a much easier problem. This is possible as the ranking function f_p can be defined in terms of the scoring function f_l:

$$f_p(\mathbf{v}_l, \mathbf{v}_k) = \text{sign}(f_l(\mathbf{v}_l) - f_l(\mathbf{v}_k)), \tag{7.4}$$

where $\text{sign}(\cdot)$ only takes the sign of the input. Plugging the scoring function model into the above equation we obtain the following:

$$\begin{aligned} f_p(\mathbf{v}_l, \mathbf{v}_k) = & \operatorname{sign}(f_l(\mathbf{v}_l) - f_l(\mathbf{v}_k)) \\ & \operatorname{sign}(\mathbf{w}^\top \mathbf{v}_l + b - \mathbf{w}^\top \mathbf{v}_k - b) \\ & \operatorname{sign}(\mathbf{w}^\top (\mathbf{v}_l - \mathbf{v}_k)) \\ & \operatorname{sign}(\mathbf{w}^\top \mathbf{z}), \end{aligned} \tag{7.5}$$

where $\mathbf{z} \in \mathbb{R}^d$ is the new descriptor extracted via: $\mathbf{z} = \mathbf{v}_l - \mathbf{v}_k$. Notice that both f_l and f_p share the same model parameter $\mathbf{w}$. As we only focus on the ranking for MULR problem, the bias parameter, b in Eq. (7.3) can be excluded; the regression model thus becomes:

$$f_l(\mathbf{v}) = \mathbf{w}^\top \mathbf{v}. \tag{7.6}$$

With the above modification, we only need to perform the training step once for both functions. To this end, we opt to perform the training step for the ranking function, f_p. Following the training formulation from the RankSVM described in [72]:

$$\frac{1}{2}\|\mathbf{w}\|_2^2 + C \sum_{m=1}^{M} \sum_{\mathbf{v}_l^{(m)}, \mathbf{v}_k^{(m)} \in \mathcal{S}_m} \ell(y_{lk}^{(m)} \mathbf{w}^\top \mathbf{z}_{lk}^{(m)}), \tag{7.7}$$

where $\mathbf{z}_{lk}^{(m)} = \mathbf{v}_l^{(m)} - \mathbf{v}_k^{(m)}$, is the new descriptor as described above; $y_{lk}^{(m)}$ is the ground truth for the MUPR problem described in Eq. (7.2), C is the training parameter and $\ell(\cdot)$ is the hinge loss.

7.4 Miss Universe (MU) Dataset

In this work, we propose the Miss Universe Dataset to address our problems. In particular, we have collected a novel dataset of videos depicting the evening gown competition for 10 years of Miss Universe (MU). The videos span from 1996 to 2010, where the judges scores are available. The videos were downloaded from YouTube and the scores were obtained from the videos themselves or Wikipedia. Fig. 7.6 shows examples of scores. While the scores taken from the videos include each individual score from judge, only the average is used (circled in yellow).

We have collected 105 videos, 18,343 frames in total, an average of 174 per video. Each video shows a candidate during the evening gown competition. Additionally, we manually select the bounding box enclosing each participant.

It is noteworthy to mention that the proposed MU dataset is extremely challenging due to variations in capture conditions for each year: (1) catwalk stage; (2) illumination conditions; (3) cameras capturing the event. As for the variations in cameras capturing the event, for our purpose we opted to use only one camera view depicting the longest walk without interruptions. Fig. 7.8 shows the catwalk stage for each year in the MU dataset. The dataset will be available for download from `http://www.itee.uq.edu.au/sas/datasets`.

Delegate [hide]	Evening Gown	Swimsuit	Average
Russia	9.64 (1)	9.88 (1) (*)	9.760 (1)
Venezuela	8.83 (5)	9.29 (2)	9.060 (2)
China	9.15 (2)	8.88 (5)	9.015 (3)
Panama	8.92 (3)	8.79 (7)	8.855 (4)
South Africa	8.79 (6)	8.90 (4)	8.845 (5)
Germany	8.84 (4)	8.81 (6)	8.825 (6)
Cyprus	8.49 (8)	9.15 (3)	8.820 (7)
Albania	8.51 (7)	8.34 (8)	8.425 (8)
India	8.10 (10)	8.32 (9)	8.210 (9)
Canada	8.39 (9)	7.99 (10)	8.190 (10)

Figure 7.6: Judges' scores. Top: taken from Wikipedia. Bottom: taken from the video.

7.4.1 Evaluation Protocol

We use leave-one-year-out protocol as the evaluation protocol for both MUPR and MULR. In particular, for each training-test set, we consider all participants from one year as the testing and the rests as training. As the dataset covers ten years of Miss Universe videos, then there will be ten training-test sets. Once the results from all the ten training-test sets are determined, the performance of a method is reported as the average of these results.

The MULR problem evaluation metric:

In the MULR problem we are interested in evaluating how similar is the ranking determined to the scoring function f_l from the actual ranking of each year. To this end, we use the Normalized Discount Cumulative Gain (NDCG) proposed to measure ranking quality of documents [89, 125]. NDCG is often used to measure of the efficacy of web search algorithms [89]. To use this metric, we consider each candidate video as a "visual" document. Here the rating of each visual document corresponds to the rank of the participant. Thus, we rate each visual document/participant video by assigning values between 1 to 10 with 10 being the highest score and 1 for the lowest, when the number of participants is 10. When the number of participants is 15, the range is between 1 to 15.

These values are assigned according to their corresponding rank. For instance, we assign the participant having the highest score with value 10 and assign the runner up with value 9. In the original formulation, the NDCG measures the ranking quality based on the top b rated documents [89]:

$$\text{NDCG}_{@b} = \frac{\text{DCG}_{@b}}{\text{IDCG}_{@b}}, \tag{7.8}$$

where DCG is the discounted cumulative gain at particular rank position b and is defined as:

$$\text{DCG}_{@b} = \sum_{j=1}^{b} \frac{(2^{r(j)} - 1)}{\log_2(\max(2, j))} \tag{7.9}$$

The rating of the $j-$th participant in the ranking list is given by $r(j)$ and $\text{IDCG}_{@b}$ is the ideal DCG at position b. Note that $b = 1, \cdots, N_q$ with N_q being the length of the ordering. A perfect list gets a $\text{NDCG}_{@b}$ score of 1. For our case, we always set $b = N_q$. We report the average percentage $\text{NDCG}_{@Nq}$ over all partitions and refer to it as the NDCG.

The MUPR problem evaluation metric:

For the MUPR problem, we use the modified Kendall's $\mathcal{K}_\tau$ as a performance measure discussed in [71]. The Kendall's $\mathcal{K}_\tau$ is defined as the number C of concordant pairs and the number D of discordant pairs. A pair $(p_l^{(m)}, p_k^{(m)})$ with $l \neq k$ is concordant, if $\hat{y}_{lk}^{(m)} = y_{lk}^{(m)}$. It is discordant if they disagree. The sum of C and D must be $\binom{N_q}{2}$. Kendall's K_τ can be defined as:

$$\mathcal{K}_\tau = \frac{C - D}{C + D} = 1 - \frac{2D}{\binom{N_q}{2}} \tag{7.10}$$

7.5 Experiments

To the best of our knowledge, this is the first work to study catwalk analysis for Miss Universe. We used the new Miss Universe dataset containing 10 versions of Miss Universe. Miss Universe 2003 contains 15 participants. The remaining versions each contain 10 participants. We used the bounding box enclosing the participant provided with the dataset. We resized all bounding boxes to 100×50.

7.5.1 Setup

All videos were converted into gray-scale. We use the leave-one-year-out protocol, where we leave one version of Miss Universe out for testing. For each video, we extract a set of $d = 14$ dimensional features as explained in Section 3.1. Empirically, we set $\beta = 40$, where β is the threshold used for selecting interesting low-level features. Parameters for the visual vocabulary GMM were learned using a large set of descriptors randomly obtained from training videos using the iterative Expectation-Maximisation algorithm [18]. Experiments were performed with three separate GMMs with varying number of components $K = \{256, 512, 1024\}$.

For the traditional FV representation, each video is represented by a FV. The FVs are fed to a linear SVM for classification. FV is our baseline system.

For the first layer of SFV, we obtained a varying number of vectors using the traditional FV representation. Each vector is obtained using the low-level descriptors of 5 consecutive- frames. Then, we advanced by a frame and obtained a new FV. For the second layer of SFV, we reduced the dimensionality of each vector from layer 1 using two methods: Principal Component Analysis (PCA) [73] and Random Projection (RP) [16].

For PCA, we retained the 90% of the energy [12]. For RP, we used the resulting dimensionality number obtained by PCA. We referred to these methods as SFV-PCA and SFV-RP.

Our classification model is described in Section 7.3.2. As explained, we address both problems using the same framework. We solve MULR by addressing MUPR first. For MUPR, we compare two catwalks belonging to the same year. The total number of comparisons is given by $\binom{N_q}{2}$. In our implementation, we solve MUPR by using the libLinear package [43] and set the bias parameter b to 0.

7.5.2 Results for MUPR

In Table 7.1, we present the results for MUPR. The evaluation metric employed is Kendall's $\mathcal{K}_\tau$ as per Eq. (7.10).

Table 7.1: Results for MUPR using $\mathcal{K}_\tau$

Method	Visual Vocabulary Size		
	256	512	1024
FV (baseline)	$52.63\ \%$	$52.16\ \%$	$52.03\ \%$
SFV-PCA	$\mathbf{56.73}\ \%$	$51.81\ \%$	$50.98\ \%$
SFV-RP	$53.68\ \%$	$46.92\ \%$	$47.87\ \%$

From this table, we can see that our classification models using both dimensionallity reduction techniques outperform the baseline FV representation. Using SFV-PCA with a visual dictionary size of 256 Gaussians leads to the best performance of 56.73%, which is 3.05 points higher than SFV-RP. PCA is an essential step for dimensionality reduction for this application.

While PCA selects the best basis vectors analysing the directions where the original data is more variable, RP selects the directions randomly. Despite the simplicity offers by random projection, the performance is still inferior than the PCA.

7.5.3 Results for MULR

Using the best setting for MUPR obtained with a visual vocabulary size of 256 Gaussians, we evaluated MULR. We also provide different random listings (RL), following the same leave-one-year-out protocol. The evaluation metric employed is NDCG as per Eq. (7.8). For random performance we iterate this procedure ten times and report the average over all iterations. Fig. 7.7 shows that SFV-PCA attained the best performance with 66.05%.

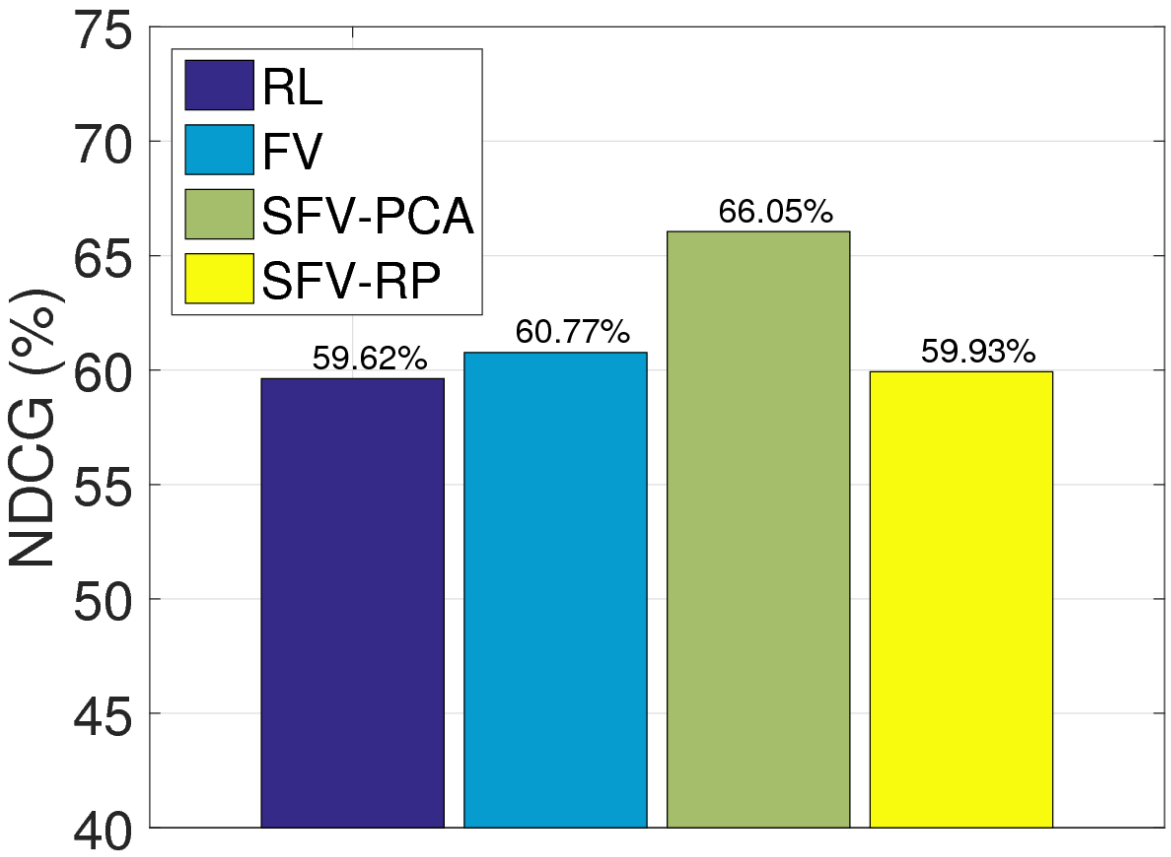


Figure 7.7: Results for MULR using NDCG.

Table 7.2 shows the individual performance using NDCG for each of the ten training-test sets as explained. Our SFV-PCA classification approach shows a performance which is higher than 50% in 7 out 10 training/test sets. In 2 out of the 10 training/test sets we obtained a performance higher than 82%. Moreover, our Miss Universe automatic prediction system was able to recognise the winner for the evening gown competition for years 1998 and 1999, which explains the higher performance for those years as in NDCG top ranked instances are considered more important. The predicted winner is also found in the top 3 for 5 out 10 versions of Miss Universe (2010, 2007, 1999, 1998, and 1996).

Table 7.2: NDCG for each year using best settings for SFV-PCA.

Year	NDCG
2010	**77.71** %
2007	**78.95** %
2003	**52.97** %
2002	**62.78** %
2001	44.37 %
2000	44.71 %
1999	**82.93** %
1998	**87.02** %
1997	51.69 %
1996	**77.38** %

7.6 Conclusions

In this work, we have present a promising approach to automatically detect the winner during the evening gown competition of Miss Universe. To this end, we have created a new dataset comprising 10 years of the evening gown competition selected from 1996 to 2010. We addressed this problem using action analysis techniques. We defined two problems that are of potential interest for the beauty pageant industry and the fashion industry. In the former problem, we are interested in predicting the winner of the competition, which can be also of interest for specialised betting sites. The fashion industry can have an innovative automatic system to compare two catwalks that can be used as training system for amateur models. Our system for predicting the winner of the evening gown competition shows we are able to rank the winner in the top 3 best predicted scores in 50% of the cases.


1996


1997


1998

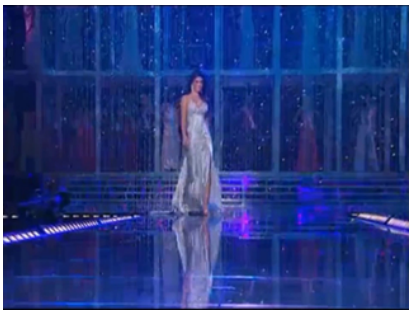

1999


2000

2001


2002


2003


2007

2010

Figure 7.8: Catwalk stages for all years.

Part III

Video Summarisation

Chapter 8

Literature Review

Many women assume they can't be good mothers and have challenging careers at the same time, so they might give up trying to do both as they get to a crucial point in their career. Although it can be hard at times, it's important for women to recognize the benefits of working outside the home.

Susan Wojcicki

The digital video is evolving fast and is bringing alone the need of new applications. It is becoming indispensable to reduce the costs of archiving, cataloguing and indexing videos [69, 157]. Several methods have been proposed to deal with these demands. Video abstraction is one of the most essential methods conceived to enhance the efficiency and manageability of stored video [69].

Video abstraction aims at providing concise representations of long videos. Video abstraction helps to quickly scan a large video database in order to efficiently access its content. It has applications in browsing and retrieval of large volumes of videos [8] and also in improving the effectiveness and efficiency of video storage [157]. Video abstraction can be categorised into two general groups: video summarisation and video skimming [69, 157].

Video summarisation, also known as still image abstraction, static storyboard or static video abstract, is a compilation of representative frames selected from the original video [39]. Video skimming, also known as moving image abstraction or moving/dynamic storyboard, is a collection of short video clips [11, 69].

Both approaches should preserve the most important content from the video in order to present a comprehensible and understandable description for the end user. In general, video skimming provides a more coherent and visually attractive result. It often retains a high-level of linguistic meaning due to its capacity to combine audio and moving elements [110, 157]. However, video summarisation is easier to generate and is not constrained in terms of timing and synchronisation [11, 157].

Video summarisation is an active area of research within the computer vision community and it has been applied in various video categories such as Wildlife Videos [181], sports videos [113], TV

documentaries [11], among others. In [8] the various approaches to video summarisation are divided into six techniques consisting of: feature selection, clustering algorithms, event detection methods, shot selection, trajectory analysis and the use of mosaics. Often a combination of techniques is used, for example one of the most common approaches is to combine feature selection with a form of clustering [11, 39, 103].

In [184] a video summary is obtained by extracting a feature vector from each frame and then clustering the resulting set of feature vectors. The smallest clusters are then removed. A keyframe – a frame that forms part of the video summary – is selected for each cluster centroid by taking the frame whose feature vector is closest to the centroid. Similar approaches are adopted in [11, 39, 41, 69] where the major difference is in the choice of feature vector used to represent each frame. Colour histograms are used in [11, 39], motion-based features are used in [41], and saliency maps are used in [69]. Each of the previously proposed feature vectors has its drawbacks. For instance, the colour histogram approach used in [11, 39] retains only coarse information about the frame. Motion-based features of [41] fail when the motion in the videos is too large. Finally, the saliency maps used in [69] perform poorly for cluttered and textured backgrounds. To date, limited work has been done on incorporating texture information to perform video summarisation.

Chapter 9

Summarisation of Short-Term and Long-Term Videos

I always did something I was a little not ready to do. I think that's how you grow. When there's that moment of 'Wow, I'm not really sure I can do this,' and you push through those moments, that's when you have a breakthrough.

Marissa Mayer

This chapter[1] presents a novel approach to video summarisation that makes use of a Bag-of-visual-Textures (BoT) approach. Two systems are proposed, one based solely on the BoT approach and another which exploits both colour information and BoT features. On 50 short-term videos from the Open Video Project we show that our BoT and fusion systems both achieve state-of-the-art performance, obtaining an average F-measure of 0.83 and 0.86 respectively, a relative improvement of 9% and 13% when compared to the previous state-of-the-art. When applied to a new underwater surveillance dataset containing 33 long-term videos, the proposed system reduces the amount of footage by a factor of 27, with only minor degradation in the information content. This order of magnitude reduction in video data represents significant savings in terms of time and potential labour cost when manually reviewing such footage.

9.1 Introduction

This chapter presents the use of texture information to improve video summarisation. We propose the use of the computationally efficient and effective bag-of-textures approach; we conjecture that this will improve video summarisation as it has been successfully applied to a range of image processing tasks, such as matching and classification of natural scenes and faces [93, 137, 180]. The bag-of-

[1]The work presented in this chapter has been published in [25].

textures model divides an image into small patches, extracts appearance descriptors from each patch, quantises each descriptor into a discrete "visual word", and then computes a compact histogram representation [52], providing considerably different information than colour histograms. In addition, we propose a fusion based system for video summarisation, where both colour and texture information is exploited. This will allow us to overcome the shortcomings of either approach. Similar approaches have been shown to be advantageous in object classification tasks [91]. We show that our system may be applied not only to short-term videos but also to long-term videos, helping in the detection of the existence of a rare species of fish.

The layout of this paper is as follows. In Section 9.2 we describe in detail our proposed video summarisation method that exploits the benefits of using texture histograms based on the bag-of-textures model. In Section 9.3 we present our improved video summarisation method that fuses the visual information provided by both the colour and texture histograms. In Section 9.4 we describe how we evaluate the video summaries of short-term and long-term videos. In Section 9.5, we present experiments which show that the proposed methods obtain higher performance than existing methods based on colour histograms. Section 9.6 summarises the main findings.

9.2 Bag-of-Textures for Video Summarisation

This section describes our proposed bag-of-textures (BoT) approach. There are four main stages:

1. Pre-processing: The input video is sub-sampled after which each frame is filtered and rescaled.
2. BoT representation:

   (i) *Local Texture Features*. Each frame is divided into small patches (blocks) and from each block we extract 2D-DCT features, which is an effective and compact representation [118].

   (ii) *Dictionary Training*. A generic visual dictionary is trained to describe the most commonly occurring textures in an independent training set.

   (iii) *Generation of BoT Histogram*. Each frame is represented by a histogram which is obtained by matching the feature vectors from each block to the dictionary.

3. Keyframe selection: Similar frames are grouped into an automatically determined number of clusters. One keyframe is selected per cluster.
4. Post-processing: In this final stage, we eliminate possible repetitive frames and create the static video summary.

Each of these stages is elucidated in the following sections.

9.2.1 Pre-processing

Sampling and Rescaling

The original input video is re-sampled to one frame per second in order to reduce the number of video frames to be examined. Each frame is then converted into gray-scale and re-scaled to be a quarter of its original size, in order to reduce the computational cost of the following stages.

Noise Filtering

There are often uninformative frames that appear at the beginning and/or the end of a segment that may affect the appearance of a video summary [39]. These frames are usually colour-homogeneous due to fade-in and fade-out effects, and have a small standard deviation of their pixel values. Frames with a standard deviation below a threshold are eliminated.

9.2.2 BoT Representation

Local Texture Features

Each frame is divided into N overlapping blocks. To each block we apply the 2D discrete cosine transform (2D-DCT) to obtain a D-dimensional feature vector that represents the local texture information [118]. Thus, the local texture feature for the n-th block of the i-th frame is $\mathbf{x}_{i,n}$.

Dictionary Training

The dictionary is trained using the *k*-means algorithm [18] by pooling the local texture features from a set of training frames. The resulting G cluster centers $\{\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_G\}$ represent the local textures (codewords) of the dictionary.

Generation of BoT Histogram

In the BoT approach the i-th frame is represented by a histogram, $\mathbf{h}_i^{\text{BoT}}$. This G-dimensional histogram represents the relative frequency of the local texture features within the frame. The g-th dimension of $\mathbf{h}_i^{\text{BoT}}$ is the relative frequency of the g-th local texture feature from the dictionary, similar to [38]. The histogram is normalised to sum to one. Thus, each local texture feature can be converted to a local histogram, $\mathbf{h}_{i,n}^{\text{BoT}}$, of dimension G where each dimension g is given by,

$$h_{g,i,n}^{\text{BoT}} = \begin{cases} 1 & \text{if } g = \underset{k \in 1,\cdots,G}{\arg\min} \|\mathbf{x}_{i,n} - \boldsymbol{\mu}_k\|_2 \\ 0 & \text{otherwise} \end{cases}. \tag{9.1}$$

These N local histograms can then be summed and normalised to produce the final BoT histogram,

$$\mathbf{h}_i^{\text{BoT}} = \frac{1}{N} \sum\nolimits_{n=1}^{N} \mathbf{h}_{i,n}^{\text{BoT}}. \tag{9.2}$$

9.2.3 Keyframe Selection

To obtain a set of keyframes we adopt an approach similar to that of [39]. A keyframe is a frame that forms part of the video summarisation. The *k*-means algorithm is used to cluster similar frames into K segments, and the resultant centroids are then used to select the keyframes.

Initially, the frames are grouped consecutively, assuming that sequential frames share similar content. To automatically determine the number of clusters, K, we calculate the Euclidean distance between two consecutive frames. If the distance is greater than a threshold τ then K is incremented. For each cluster centroid the frame whose BoT histogram is closest is selected as a keyframe. A total of K keyframes is then reached.

9.2.4 Post-processing

Having obtained the initial set of K keyframes we then attempt to discard those keyframes which are too similar. This is achieved by comparing all keyframes against each other. If the Euclidean distance between the BoT histograms of the keyframes is smaller than a threshold τ then one of the two keyframes under consideration is discarded. This gives the final static video summary that consists of N_{as} keyframes, where $N_{as} \leq K$, with as standing for automatic summary.

Lastly, the static video summary is obtained after organising the resulting keyframes in temporal order.

9.3 Fusion of Colour and BoT

In this section, we present a hybrid system that fuses colour histograms [39] and BoT texture information, termed as CaT (for **C**olour **a**nd **T**exture). The proposed CaT approach to video summarisation has the same 4 stages as our proposed BoT video summarisation approach, but with additions in order to obtain colour histograms. We describe these additions below.

1. Pre-processing: The input video is processed in two independent ways. First, we obtain the BoT histograms as described in Section 9.2.1. Second, to obtain the colour histograms we extract the Hue component, from the HSV colour space, of the unscaled input frame similar to [39]. In both cases we remove uninformative frames by employing the noise filtering process described in Section 9.2.1.
2. Texture and Colour Histogram: The BoT histogram is the same as explained in Section 9.2.2. The colour histogram, $\mathbf{h}_i^{\text{hue}}$, of the i-th frame is computed using only the Hue component as in [39].
3. Keyframe Selection: The BoT and colour histograms are clustered using k-means. This stage is similar to Section 9.2.3. The difference lies in the distance measure used to compare all frames against each other.

(i) To select the number of keyframes K we combine the information from the BoT and colour histograms. When calculating the distance between frame a and b we use the weighted summation of Euclidean distances:

$$\alpha\|\mathbf{h}_a^{\text{BoT}} - \mathbf{h}_b^{\text{BoT}}\|_2 + \beta\|\mathbf{h}_a^{\text{hue}} - \mathbf{h}_b^{\text{hue}}\|_2 \tag{9.3}$$

under the constraints $\alpha + \beta = 1$, $\alpha \geq 0$, $\beta \geq 0$.

(ii) Each keyframe is selected by finding the frame which is closest to each cluster centroid. For the CaT approach the distance between a frame and a centroid is calculated as a weighted summation of the Euclidean distances, as per Eq. (9.3).

4. Post-processing: To eliminate similar frames we use the procedure described in Section 9.2.4 but replace the Euclidean distance with the weighted summation of the Euclidean distances, as per Eq. (9.3).

9.4 Datasets and Evaluation Metrics

To evaluate the performance of video summarisation we use two datasets consisting of short- and long-term video data. The short-term data is obtained from the Open Video Project[2]. The long-term data is a new dataset that consists of 14 hours of underwater video surveillance which monitors the behaviour of marine wildlife.

9.4.1 Short-Term Videos

We use the 50 videos from the Open Video Project which contain ground truth [39]. Each ground truth consists of the summary provided by $P = 5$ users. The users provided the summaries under no restrictions upon length nor appearance of the summaries.

To evaluate the performance on the short-term video data we use the "Comparison of User Summaries" (CUS) method [39]. This method compares the automatic video summarisation and ground truth by exhaustively calculating the distance between the frames from the automatic summarisation and the ground truth. Two frames are similar if the distance between their respective feature vectors (histograms) is less than an evaluation threshold δ. If the frames match they are removed from the next iteration of the comparison process. For performance evaluation, the distance measure used for the BoT approach is the Euclidean distance, however, to be consistent with prior work [39], the distance measure for the colour histograms is the L_1-norm. Therefore, the distance measure used for CaT is the weighted summation of the Euclidean distance for the BoT histograms and the L_1-norm for the colour histograms:

$$\alpha\|\mathbf{h}_a^{\text{bof}} - \mathbf{h}_b^{\text{bof}}\|_2 + \beta\|\mathbf{h}_a^{\text{hue}} - \mathbf{h}_b^{\text{hue}}\|_1. \tag{9.4}$$

[2]Open Video Project: *http://www.open-video.org*

Various evaluation metrics exist to measure the quality of an automatic video summary. We use three evaluation metrics so that we can compare our proposed approaches with two state-of-the-art methods [39, 11]. To compare with [39] we use accuracy (acc) and error (err), and to compare with [11] we use the F-measure.

To calculate acc and err, each frame in the automatic video summary is compared with all frames in the user summary and then the number of matching frames (N_m) and non-matching frames (N_{nm}) are calculated:

$$acc = \frac{N_m}{N_u}, \quad err = \frac{N_{nm}}{N_u} \tag{9.5}$$

where N_{as} and N_u are the total number of frames from the automatic and user summary, respectively.

The F-measure, defined as

$$F = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} \ + \ \text{recall}} \tag{9.6}$$

is used to to provide a single number that balances precision $= N_m/N_{as}$ and recall $= N_m/N_u$.

The evaluation metrics are presented as an average. First, we take the average from the P users to obtain acc_P, err_P, and F_P; for each video there are $P = 5$ users. Then we take the average across all of the videos to obtain $\overline{acc}$, $\overline{err}$, and $\overline{F}$. In terms of $\overline{acc}$ it is desirable to have a high value as it measures the number of matching frames. In terms of $\overline{err}$ it is desirable to have a small value as it measures the number of non-matching frames. With regards to $\overline{F}$ it is desirable to obtain a high value, which occurs when the precision and recall are large.

9.4.2 Long-Term Videos

The long-term videos consist of 14 hours of underwater footage from 33 videos which are on average 25 minutes in duration. This data was obtained from the NSW-DPI[3], courtesy of David Harasti. Example images are shown in Figure 9.1. In each video there is always at least one segment where a rare species of fish, the black cod, is within view. Normally these videos would be inspected by a human expert to determine if there is an instance of the rare fish within. We propose that video summarisation can be used to reduce the amount of footage to be viewed in order to detect the existence of this rare species of fish.

Using ground truth which provides time-stamps when this rare species is within view, we examine the effectiveness of video summarisation to provide at least one keyframe in each static video summary with the rare species of interest within view. This is useful as it presents a way to reduce the time and cost of manually viewing a large amount of video data.

To calculate the performance of long-term videos we present results in terms of detection accuracy and the average compression ratio (R_c). Detection accuracy refers to whether an instance of the rare species is among any of the chosen keyframes for a static video summary; 75% would mean that there is at least 1 keyframe of the rare species in 75% of the static video summaries.

[3]New South Wales Department of Primary Industries, Australia.

Figure 9.1: Example images from the long-term underwater surveillance videos; the added red ellipsoids highlight the rare species of interest.

To calculate the average compression ratio we first note that because we have long-term videos then for each video there might be many hundreds of keyframes. To present all of these keyframes effectively to the user we re-encode them into a static video summary by presenting each keyframe for 0.25 seconds. This gives the user time to effectively view the keyframe. Thus the t-th long-term video $\mathbf{V}_t$ is converted to a static video summary $\mathbf{S}_t$ with a compression ratio given by:

$$R_{c,t} = 4 \times \frac{\text{Duration}(\mathbf{V}_t)}{\text{Duration}(\mathbf{S}_t)} \tag{9.7}$$

where Duration is the duration of a video and the factor of 4 is introduced as there are 4 keyframes per second of the shortened video.

9.5 Experiments

An important part of both the BoT and CaT approaches is the training of the dictionary to obtain the texture histograms. To train this dictionary we use 10 frames randomly selected from videos taken from the Open Video Project that have no user summaries, ensuring they are independent of the evaluation dataset. In addition, the frames selected to train the dictionary look significantly different to the ground truth provided by the users.

To obtain the proposed local texture features we divide each frame into a set of overlapping blocks. Similar to [137] we use a block size of 8×8 with an overlap margin of 6 pixels, and represent each block as a $D = 15$ dimensional feature vector containing 2D-DCT coefficients. We extract the first 16 2D-DCT coefficients, which represent low-frequency information [118], and omit the first coefficient as it is the most sensitive to illumination changes. With regards to the colour histogram, we quantise the Hue component into 16 bins as per [39]. These parameters are the same for all experiments.

The values for the threshold τ, fusion weight α and evaluation threshold δ were determined experimentally. For all of the experiments we search for the optimal fusion parameter $\alpha = \{0.0, 0.1, \cdots, 1.0\}$. Our proposed methods were implemented using the OpenCV [20] and Armadillo [136] C++ libraries.

9.5.1 Short-Term Videos

We compare the performance against two baseline systems from literature: *(i)* VSUMM [39] and *(ii)* VISON [11]. The two baseline systems use colour information as their primary feature. VSUMM uses colour information by retaining only the Hue component of HSV and generating a histogram of 16 bins. VISON is a state-of-the-art approach and consists of a histogram of the HSV representation of each frame. It combines the HSV information in a compressed form such that the Hue component is treated with greater importance and results in a histogram of 256 bins.

An initial set of experiments were performed to find the optimal number of components for the dictionary of our proposed texture features. Using a fixed number of components $G = \{8, 16, 32\}$ and a fixed number of thresholds $\tau = \{0.05, 0.10, \cdots, 0.5\}$, we found that using just $G = 8$ components provided optimal performance. We kept the number of components constant for the remainder of our experiments.

In Figure 9.2 we present a summary of the average performance for 50 short-videos of our proposed systems, BoT and CaT, and the two baselines. Two interesting results can be seen from this figure.

First, it can be seen that the texture-only BoT system performs better than either the VSUMM or VISON approaches which primarily use colour information. The BoT system obtains an average F-measure of $\overline{F} = 0.83$, which is a relative improvement of 9% when compared to VISON, $\overline{F} = 0.76$. Furthermore, the $\overline{acc}$ and $\overline{err}$ of the BoT system shows that it produces a more accurate summarisation than VSUMM and also has the lowest $\overline{err}$ of any system[4]. This suggests that texture information is either equally or more important than colour information for the task of video summarisation.

Second, the proposed CaT system (fusing colour histograms and the proposed texture histograms) performs better than the two baseline systems and the proposed texture-only BoT system. The CaT system has an average F-measure of $\overline{F} = 0.86$, which is a relative improvement of 13% when compared to VISON $\overline{F} = 0.76$, the previous state-of-the-art approach.

Figure 9.3 shows the qualitative results for the automatic summarisation provided by VSUMM and VISON as well as our proposed BoT and CaT systems. It can be seen that VSUMM (Figure 9.3a) with $F_P = 0.83$, VISON (Figure 9.3b) with $F_P = 0.78$, and our proposed BoT (Figure 9.3c) with $F_P = 0.74$ contain some keyframes that may not be of interest and/or are repetitive. In contrast, the proposed CaT system (Figure 9.3d) provides the most consistent video summary with $F_P = 0.86$.

9.5.2 Long-Term Videos

In this section we present results on 33 long-term videos which last on average for 25 minutes. We examine the applicability of video summarisation to long-term videos to efficiently detect a rare species of fish and measure performance in terms of detection accuracy and compression rate (see Section 9.4.2).

[4]No results in terms of $\overline{acc}$ and $\overline{err}$ were supplied for VISON in [11].

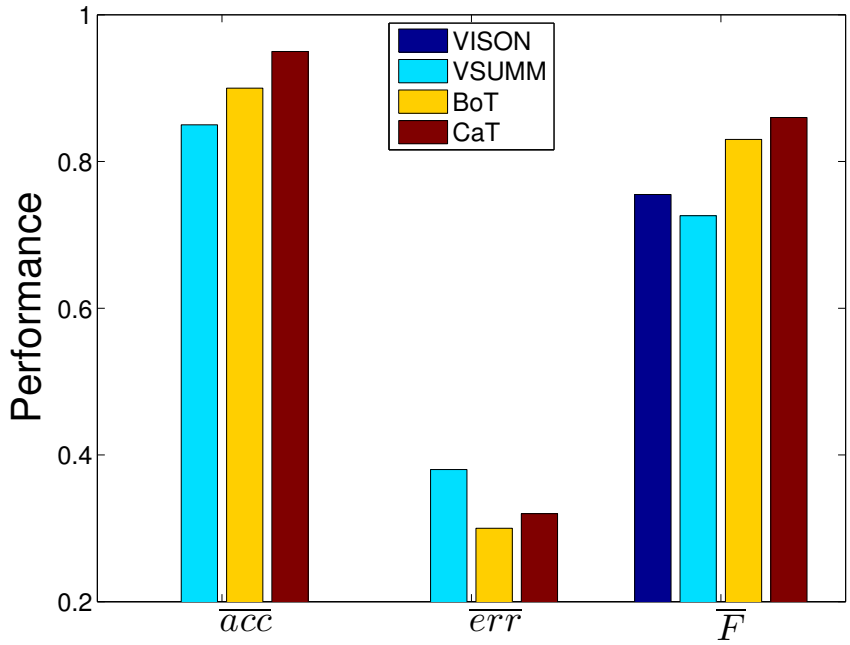


Figure 9.2: Comparative evaluation of our proposed methods with VSUMM [39] and VISON [11]. Lower values of $\overline{err}$ as well as higher values of $\overline{acc}$ and $\overline{F}$ are desired.

The accuracy and average compression ratio of the algorithm for various thresholds:

$$\tau = \{0.025, 0.05, \ldots, 0.1\},$$

is presented in Figure 9.4. It can be seen in Figure 9.4a that the CaT algorithm consistently outperforms the BoT and VSUMM algorithms. We attribute this to the fact that the background in these videos is relatively stable and so the colour histograms used in VSUMM do not change as often compared to the short-term videos used in [39].

In Figure 9.4b it can be seen that while using the VSUMM algorithm provides better average compression ratio than either the BoT or CaT approaches, it comes at the cost of accuracy. In general the proposed fusion approach provides the most consistent trade-off between accuracy and average compression ratio.

We take the optimal system at the threshold $\tau = 0.05$ as this provides a high degree of detection accuracy, 85%, and a good average compression ratio of 27. This system will allow a user to see the fish of interest in 85% of the summarised videos while reducing the amount of video data to view by 27 times, more than an order of magnitude. Such an approach would reduce the 14 hours of video data to just 31 minutes, thus enabling significantly more efficient reviewing of the data.

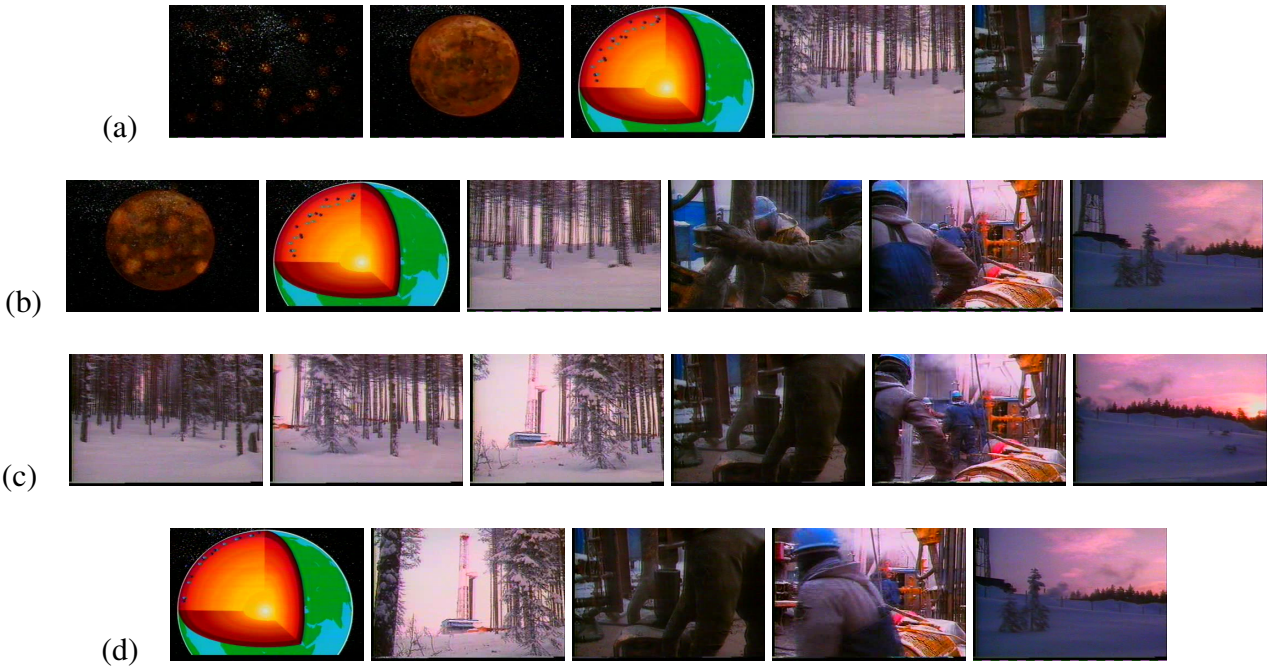


Figure 9.3: Static video summary for *"the future of energy gases - segment 09"*, using (a) VSUMM, (b) VISON, (c) proposed BoT, and (d) proposed CaT.

9.6 Conclusions

In this chapter, we have proposed the novel use of textures to perform video summarisation. We proposed to use a visual-bag-of-textures (BoT) in two ways. First, a BoT system which uses only texture features is proposed and it is shown to outperform two state-of-the-art systems which use colour only, VSUMM and VISON. Second, a fused system that combines Colour and Texture (CaT) is proposed and it is shown to provide further improvements.

Both of our proposed systems outperform two state-of-the-art approaches, VSUMM and VISON, which use colour features. Experiments on 50 short-term videos, obtained from the Open Video Project, show that our proposed texture-only system (BoT) obtains an F-measure of 0.83, which is better than either VSUMM or VISON which obtain an average F-measure of 0.73 and 0.76, respectively. Furthermore, our fused system (CaT) demonstrates that combining colour and texture features yields state-of-the-art performance with an average F-measure of 0.86.

We have also shown that video summarisation can be applied effectively to long-term videos. Using 33 long-term surveillance videos, in our case underwater surveillance footage, we have shown that video summarisation can be used to significantly reduce the amount of footage to view, by up to a factor of 27, with only a minor degradation in the information content.

(a)

(b)

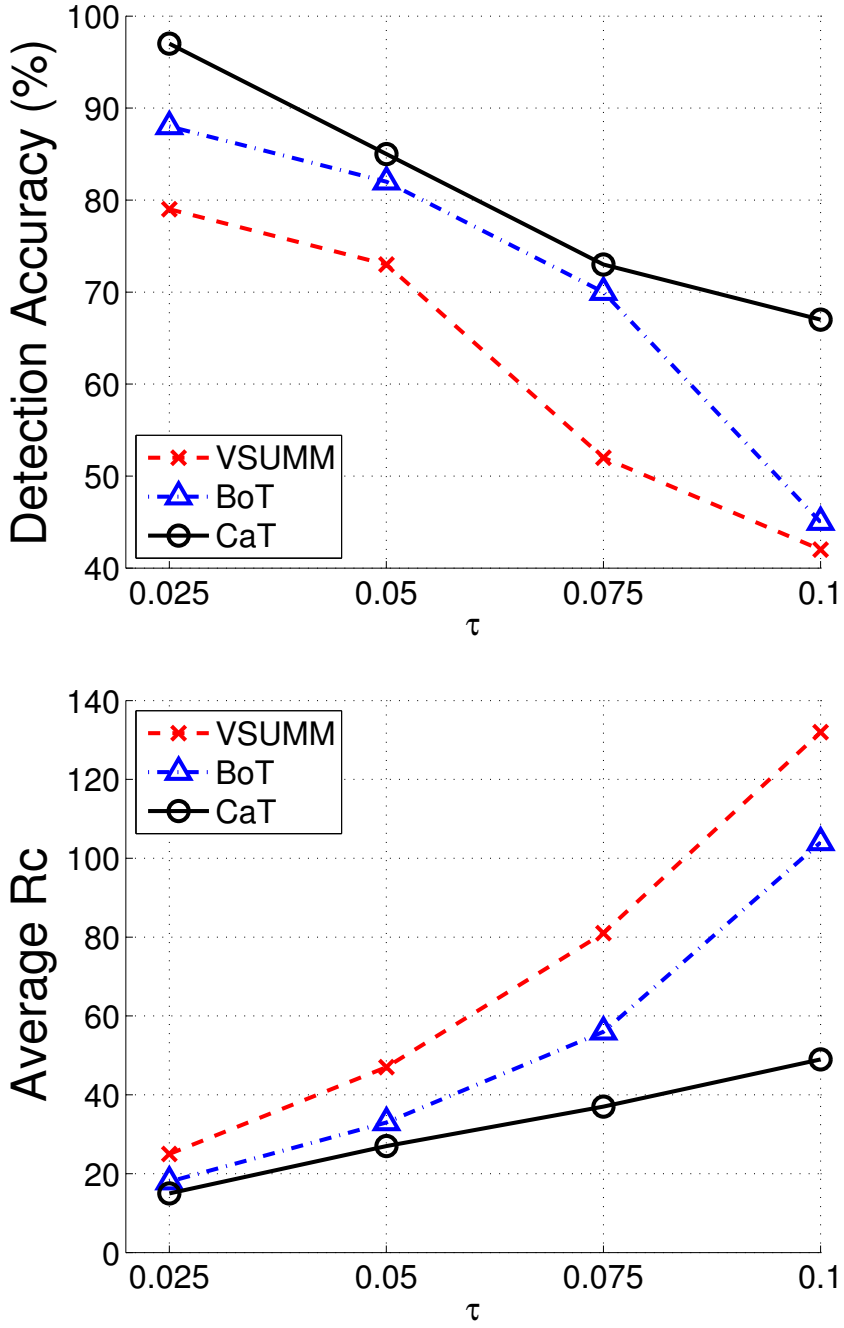


Figure 9.4: Demonstration of the trade-off between (a) the detection accuracy and (b) the average compression ratio R_c for the 33 long-term videos using the CaT, BoT and VSUMM approaches.

Part IV

Final Remarks

Chapter 10

Overall Main Findings

If you are successful, it is because somewhere, sometime, someone gave you a life or an idea that started you in the right direction. Remember also that you are indebted to life until you help some less fortunate person, just as you were helped.

Melinda Gates

In recent years, we have witnessed how video data has exponentially increased. It has been also forecast that video data will be responsible for the majority of online traffic in the next few years. Undoubtedly, it becomes necessary to develop automatic and intelligent systems to efficiently analyse, process, and interpret the information contained in the video data. This thesis contributes towards efficiently manage and interpret video information via action analysis and video summarisation. In this chapter a summary of the contributions and main findings for the topics encompassed in this thesis are provided.

10.1 Main Findings for Action Analysis

Given that action analysis is a broad topic that covers several areas, this thesis addresses the following three areas with action analysis: action recognition, joint action segmentation and recognition, and action assessment. The main findings per each topic are given below.

10.1.1 Comparative Evaluation of Action Recognition Approaches

In chapter 5, we have brought out an extensive empirical comparison among existing techniques for the human action recognition problem. Experiments have been carried out using three popular datasets: KTH, UCF-Sports and UT-Tower. We have analysed Riemannian representations including nearest-neighbour classification, kernel methods, and kernelised sparse representations.

For Riemannian representation we used covariance matrices of features, which are symmetric positive definite (SPD), as well as linear subspaces (LS). Moreover, we compared all the aforementioned Riemannian representations with GMM and FV based representations, using the same extracted features. We also evaluated the robustness of the most representative approaches to translation and scale variations. For manifold representations, all SPD matrices approaches surpass their LS counterpart. The FV representation outperforms all the techniques under ideal and challenging conditions. All techniques are affected when facing challenging conditions. However, FV is less sensitive under moderate variations in both scale and translation.

10.1.2 Joint Action Recognition and Segmentation

In chapter 6, we have proposed two hierarchical approaches that perform joint action segmentation and classification in videos: PI-FV and PI-GMM. Videos are processed through overlapping temporal windows.

For the PI-FV, features from each window are represented as a Fisher vector, which captures the first and second order statistics. Rather than directly classifying each Fisher vector, it is converted into a vector of class probabilities. For PI-GMM, the vector of class probabilities is obtained using the average log-likelihood over each temporal window. The final classification decision for each frame (action label) is then obtained by integrating the class probabilities at the frame level, which exploits the overlapping of the temporal windows. The proposed approach has a lower number of free parameters than previous methods. We have found that PI-FV it is also considerably less computationally demanding compared to modelling each action directly with PI-GMM.

Experiments were done on two datasets: s-KTH (a stitched version of the KTH dataset to simulate multi-actions), and the more challenging CMU-MMAC dataset (containing realistic multi-action videos of food preparation). On s-KTH, the proposed PI-FV considerably outperforming proposed PI-GMM and HMM-based. On CMU-MMAC, the proposed approach outperforms the PI-GMM and HMM. Furthermore, the proposed system PI-FV is much faster than the also proposed PI-GMM approach.

10.1.3 Catwalk Analysis (Action Assessment)

In chapter 7, we have presented a novel and promising approach to automatically detect the winner during the evening gown competition of Miss Universe. To this end, we have created a new dataset comprising 10 years of the evening gown competition selected from 1996 to 2010. We addressed this problem using action analysis techniques. We defined two problems that are of potential interest for the beauty pageant industry and the fashion industry. In the former problem, we are interested in predicting the winner of the competition, which can be also of interest for specialised betting sites. The fashion industry can have an innovative automatic system to compare two catwalks that can be used as training system for amateur models. Our system for predicting the winner of the evening

gown competition shows we are able to rank the winner in the top 3 best predicted scores in 50% of the cases.

10.2 Main Findings for Video Summarisation

In chapter 9, we have proposed the novel use of textures to perform video summarisation. We proposed to use a visual-bag-of-textures (BoT) in two ways. First, a BoT system which uses only texture features is proposed. Second, a fused system that combines Colour and Texture (CaT). Both of our proposed systems outperform two state-of-the-art approaches, VSUMM and VISON, which use colour features.

Experiments on 50 short-term videos, obtained from the Open Video Project, show that our proposed texture-only system (BoT) is better than two recent approaches. Furthermore, our fused system (CaT) demonstrates that combining colour and texture features yields state-of-the-art performance. We have also shown that video summarisation can be applied effectively to long-term videos. Using 33 long-term surveillance videos, in our case underwater surveillance footage, we have shown that video summarisation can be used to significantly reduce the amount of footage to view with only a minor degradation in the information content.

Chapter 11

Potential Future Work

I am thankful for the way I was raised, to be positive. Even when times have gotten rough I have always tried to look on the bright side. Even when I was put down, yelled at and made feel insignificant, I still thought things were alright. I did realise when enough is enough.

Angela Merkel

As part of future research, the proposed approaches can be extended or modified as follows.

11.1 Future Work for Action Analysis

11.1.1 Comparative Evaluation of Action Recognition Approaches

- It would be interesting to explore ways to improve the Riemannian representation in order to equal or surpass the performance of the FV approach.
- One possible way is to flatten the manifold via RKHS and then use the FV representation. This combination of Riemannian manifolds and FV will allow us to exploit the benefits of both worlds.
- As flattening the Riemannian manifolds is not a straightforward problem, we could employ a recent work which implements random projections for manifold points via kernel space while preserving the geometric structure of the original space [183].
- The RKHS is constructed from a small subset of data. All Riemannian points are then projected into a new space where traditional Euclidean techniques can be employed.

11.1.2 Joint Action Recognition and Segmentation

- The proposed system for joint action recognition and segmentation can be further sped by using the fast Fisher vector variant proposed in [147], where for each sample the deviations for only one Gaussian are calculated. This can deliver a large speed up in computation, at the cost of a small drop in accuracy [115].
- Actions like twist-off is often followed by twist-on. However, our system tends to confuse both actions. We may explore semantic relationships between actions as in [126] to increase the system performance.
- Improved dense trajectories (IDT) [168] have exhibited superior performance for single action recognition in recent years. In order to use dense trajectories for joint action recognition and segmentation, it would be necessary to overcome its limitations. Those limitations include the incapability of IDT to distinguish between objects of interest and background. This causes an unwanted and problematic increase in the costs of computations and data storage, which can be particularly critical when dealing with large dataset like CMU-MMAC.
- Another potential area of inquiry is to replace the Fisher vector representation by treating a deep convolutional neural network (DCNN) as a high-level feature extractor [128]. This in turn may require an extension of the DCNN architecture to explicitly incorporate spatio-temporal information.

11.1.3 Catwalk Analysis (Action Assessment)

- The dataset for catwalk analysis can be enlarged using other Miss Universe versions, other beauty pageant competitions, and catwalks from international fashion trade shows. Given that scores are not always publicly available, an online competitive Catwalk rating game can be designed similar to the style rating game called *Hipster Wars* [80]. With this online game it would be possible to crowd source reliable human judgements of catwalks.
- Catwalk analysis can be extended to the swimsuit catwalk competition, which together with the evening gown competition are critical to the selection of the next Miss Universe. For the swimming competition, other attributes apart from the catwalk would be needed to take into consideration such as good muscle tone, body proportion, body fat, body shape, and fitness. All those attributes are also visual attributes.
- Pose is an important attribute for catwalks. We envisage that pose-based CNN features in conjunction with IDT can increase our system performance. This combination has been latterly proposed and has shown to be effective for action recognition [34].
- This work can be also extended to other applications that requires action assessment. For instance, patient rehabilitation and high performance sports. In both cases, an automatic system able to evaluate the progress of a patient or an athlete would be valuable.

11.2 Future Work for Video Summarisation

- Future work should examine alternative features and application settings with a particular emphasis for long-term videos.
- For instance, emphasising the importance of foreground objects [130] should be explored, as well as explicit modelling of movement (or actions) of such objects [58, 138].
- The applicability of video summarisation to CCTV surveillance footage should also be considered.
- With video containing humans, summarisation and action analysis can be jointly addressed. When dealing with both at the same time as freshly proposed in [64], action recognition can extract the relevant and descriptive information to create a high level video summary. Instead of selecting key-frames, we can select key actions that best describe videos.

Bibliography

[1] "List of beauty pageants," http://en.wikipedia.org/wiki/List_of_beauty_pageants.

[2] "Miss Universe," http://www.missuniverse.com/.

[3] "Miss Universe in Wikipedia," http://en.wikipedia.org/wiki/Miss_Universe.

[4] "Cisco visual networking index: Forecast and methodology, 2014 to 2019," Cisco Systems, Inc, Tech. Rep., 2015.

[5] "Video surveillance camera installed base report," IHS Technology, Tech. Rep., 2015.

[6] J. Aggarwal and M. Ryoo, "Human activity analysis: A review," *ACM Comput. Surveys*, vol. 43, no. 3, pp. 16:1–16:43, 2011.

[7] R. K. Aggarwal, Namita; Agrawal, "First and second order statistics features for classification of magnetic resonance brain images," *Journal of Signal & Information Processing*, 2012.

[8] M. Ajmal, M. Ashraf, M. Shakir, Y. Abbas, and F. Shah, "Video summarization: techniques and classification," in *Lecture Notes in Computer Science, Vol. 7594*, 2012, pp. 1–13.

[9] A. Alavi, M. T. Harandi, and C. Sanderson, "Relational divergence based classification on Riemannian manifolds," in *Applications of Computer Vision (WACV)*, 2013, pp. 111–116.

[10] S. Ali and M. Shah, "Human action recognition in videos using kinematic features and multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 288–303, 2010.

[11] J. Almeida, N. J. Leite, and R. da S. Torres, "VISON: VIdeo Summarization for ONline applications," *Pattern Recognition Letters*, vol. 33, no. 4, pp. 397–409, 2012.

[12] X. Amatriain, A. Jaimes, N. Oliver, and J. M. Pujol, *Recommender Systems Handbook*, 2011, ch. Data Mining Methods for Recommender Systems, pp. 39–71.

[13] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Log-Euclidean metrics for fast and simple calculus on diffusion tensors," in *Magnetic Resonance in Medicine*, vol. 56, 2006, pp. 411–432.

[14] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt, "Sequential deep learning for human action recognition," in *Human Behavior Understanding*, ser. Lecture Notes in Computer Science, 2011, vol. 7065, pp. 29–39.

[15] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer Vision Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[16] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: Applications to image and text data," in *International Conference on Knowledge Discovery and Data Mining ACM SIGKDD*, ser. KDD '01. ACM, 2001, pp. 245–250.

[17] D. A. Bini and B. Iannazzo, "Computing the Karcher mean of symmetric positive definite matrices," *Linear Algebra and its Applications*, vol. 438, no. 4, pp. 1700 – 1710, 2013.

[18] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[19] E. Borzeshi, O. Perez Concha, R. Xu, and M. Piccardi, "Joint action segmentation and classification by an extended hidden Markov model," *Signal Processing Letters*, vol. 20, no. 12, pp. 1207–1210, 2013.

[20] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[21] D. Buchsbaum, K. R. Canini, and T. Griffiths, "Segmenting and recognizing human action using low-level video features," *Annual Conference of the Cognitive Science Society*, 2011.

[22] L. Cao, Y. Tian, Z. Liu, B. Yao, Z. Zhang, and T. Huang, "Action detection using multiple spatial-temporal interest point features," in *International Conference on Multimedia and Expo (ICME)*, 2010, pp. 340–345.

[23] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li, "Learning to rank: From pairwise approach to listwise approach," in *International Conference on Machine Learning*, 2007, pp. 129–136. [Online]. Available: http://doi.acm.org/10.1145/1273496.1273513

[24] J. Carvajal, C. McCool, B. C. Lovell, and C. Sanderson, "Joint recognition and segmentation of actions via probabilistic integration of spatio-temporal Fisher vectors," in *Trends and Applications in Knowledge Discovery and Data Mining*, ser. Lecture Notes in Computer Science, vol. 9794, 2016, pp. 115–127.

[25] J. Carvajal, C. McCool, and C. Sanderson, "Summarisation of short-term and long-term videos using texture and colour," in *Winter Conference on the Applications of Computer Vision (WACV)*, 2014, pp. 769–775.

[26] J. Carvajal, C. Sanderson, C. McCool, and B. C. Lovell, "Multi-action recognition via stochastic modelling of optical flow and gradients," in *Workshop on Machine Learning for Sensory Data Analysis (MLSDA)*, 2014, pp. 19–24.

[27] J. Carvajal, A. Wiliem, C. McCool, B. C. Lovell, and C. Sanderson, "Comparative evaluation of action recognition methods via Riemannian manifolds, Fisher vectors and GMMs: Ideal and challenging conditions," in *Trends and Applications in Knowledge Discovery and Data Mining*, ser. Lecture Notes in Computer Science, vol. 9794, 2016, pp. 88–100.

[28] J. Carvajal, A. Wiliem, C. Sanderson, and B. C. Lovell, "Towards Miss Universe automatic prediction: The evening gown competition," in *International Conference on Pattern Recognition (ICPR)*, 2016.

[29] O. Chapelle and S. S. Keerthi, "Efficient algorithms for ranking with SVMs," *Information Retrieval*, vol. 13, no. 3, pp. 201–215, 2009.

[30] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman, "The devil is in the details: an evaluation of recent feature encoding methods," in *British Machine Vision Conference (BMVC)*, 2011.

[31] C.-C. Chen and J. K. Aggarwal, "Recognizing human action from a far field of view," *Workshop on Motion and Video Computing (WMVC)*, pp. 1–7, 2009.

[32] W. Chen, T. Liu, Y. Lan, Z. Ma, and H. Li, "Ranking measures and loss functions in learning to rank," in *Advances in Neural Information Processing Systems 22*, 2009, pp. 315–323.

[33] Y. Cheng, Q. Fan, S. Pankanti, and A. Choudhary, "Temporal sequence modeling for video event detection," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 2235–2242.

[34] G. Chéron, I. Laptev, and C. Schmid, "P-CNN: Pose-based CNN features for action recognition," in *International Conference on Computer Vision (ICCV)*, 2015, pp. 3218–3226.

[35] K. Crammer and Y. Singer, "On the algorithmic implementation of multiclass kernel-based vector machines," *Journal of Machine Learning Research*, vol. 2, pp. 265–292, 2001.

[36] G. Csurka and F. Perronnin, "Fisher vectors: Beyond bag-of-visual-words image representations," in *Computer Vision, Imaging and Computer Graphics. Theory and Applications*, 2011, vol. 229, pp. 28–42.

[37] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *European Conference on Computer Vision (ECCV)*, ser. Lecture Notes in Computer Science, A. Leonardis, H. Bischof, and A. Pinz, Eds., 2006, vol. 3952, pp. 428–441.

[38] N. Dardas, Q. Chen, N. D. Georganas, and E. Petriu, "Hand gesture recognition using bag-of-features and multi-class support vector machine," in *IEEE Int. Symp. Haptic Audio-Visual Environments and Games (HAVE)*, 2010, pp. 1–5.

[39] S. E. F. de Avila, A. P. B. Lopes, A. da Luz Jr., and A. de Albuquerque Araújo, "VSUMM: a mechanism designed to produce static video summaries and a novel evaluation method," *Pattern Recognition Letters*, vol. 32, no. 1, pp. 56–68, 2011.

[40] F. De la Torre, J. K. Hodgins, J. Montano, and S. Valcarcel, "Detailed human data acquisition of kitchen activities: the CMU-multimodal activity database (CMU-MMAC)," in *CHI Workshop on Developing Shared Home Behavior Datasets to Advance HCI and Ubiquitous Computing Research*, 2009.

[41] A. Divakaran, K. A. Peker, and H. Sun, "Video summarization using motion descriptors," in *Proc. SPIE Conf. on Storage and Retrieval from Multimedia Databases*, 2001.

[42] M. Douze, A. Ramisa, and C. Schmid, "Combining attributes and Fisher vectors for efficient image retrieval," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 745–752.

[43] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *Journal of Machine Learning Research*, pp. 1871–1874, 2008.

[44] M. Faraki, M. T. Harandi, and F. Porikli, "More about VLAD: A leap from Euclidean to Riemannian manifolds," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[45] I. Fatima, M. Fahim, Y.-K. Lee, and S. Lee, "A unified framework for activity recognition-based behavior analysis and action prediction in smart homes," *Sensors*, vol. 13, no. 2, pp. 2682–2699, 2013.

[46] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[47] V. Fečová, L. Nováková-Marcinčinová, M. Janák, J. Novák-Marcinčin, J. Barna, and J. Török, "Devices and software possibilities for using of motion tracking systems in the virtual reality system," in *International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, 2012, pp. 165–168.

[48] A. Gaidon, Z. Harchaoui, and C. Schmid, "Actom sequence models for efficient action detection," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[49] F. Gholami, D. A. Trojan, J. Kövecses, W. M. Haddad, and B. Gholami, "Gait assessment for multiple sclerosis patients using Microsoft Kinect," vol. arXiv preprint 1508.02405, 2015.

[50] B. Gong, W.-L. Chao, K. Grauman, and F. Sha, "Diverse sequential subset selection for supervised video summarization," in *Advances in Neural Information Processing Systems 27*, 2014.

[51] D. Gong, G. Medioni, and X. Zhao, "Structured time series analysis for human action segmentation and recognition," *Pattern Analysis and Machine Intelligence*, pp. 1414–1427, 2014.

[52] K. Grauman and B. Leibe, "Visual object recognition," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 5, no. 2, pp. 1–181, 2011.

[53] O. K. Gross, Y. Gurovich, T. Hassner, and L. Wolf, "Motion interchange patterns for action recognition in unconstrained videos," in *European Conference on Computer Vision (ECCV)*, 2012.

[54] K. Guo, P. Ishwar, and J. Konrad, "Action recognition from video using feature covariance matrices," *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2479–2494, 2013.

[55] ——, "Action recognition in video by sparse representation on covariance manifolds of silhouette tunnels," in *Recognizing Patterns in Signals, Speech, Images and Videos*, 2010, vol. 6388, pp. 294–305.

[56] J. Hamm and D. D. Lee, "Grassmann discriminant analysis: A unifying view on subspace-based learning," in *International Conference on Machine Learning (ICML)*, 2008, pp. 376–383.

[57] M. Harandi, C. Sanderson, C. Shen, and B. Lovell, "Dictionary learning and sparse coding on Grassmann manifolds: An extrinsic solution," in *International Conference on Computer Vision (ICCV)*, 2013, pp. 3120–3127.

[58] M. T. Harandi, C. Sanderson, S. Shirazi, and B. C. Lovell, "Kernel analysis on Grassmann manifolds for action recognition," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 1906–1915, 2013.

[59] M. Harandi, C. Sanderson, R. Hartley, and B. C. Lovell, "Sparse coding and dictionary learning for symmetric positive definite matrices: A kernel approach," in *European Conference on Computer Vision (ECCV)*, 2012, pp. 216–229.

[60] M. Harandi, C. Sanderson, A. Wiliem, and B. Lovell, "Kernel analysis over Riemannian manifolds for visual recognition of actions, pedestrians and textures," in *Workshop on Applications of Computer Vision (WACV)*, 2012, pp. 433–439.

[61] T. Hassner, "A critical review of action recognition benchmarks," in *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2013, pp. 245–250.

[62] S. Hirose, I. Nambu, and E. Naito, "An empirical solution for over-pruning with a novel ensemble-learning method for fMRI decoding," *Journal of Neuroscience Methods*, vol. 239, pp. 238 – 245, 2015.

[63] M. Hoai, Z.-Z. Lan, and F. De la Torre, "Joint segmentation and classification of human actions in video," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 3265–3272.

[64] F. Hussein, S. Awwad, and M. Piccardi, "Joint action recognition and summarization by sub-modular inference," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 2697–2701.

[65] T. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," in *Advances in Neural Information Processing Systems 11*, 1998, pp. 487–493.

[66] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, and M. Harandi, "Optimizing over radial kernels on compact manifolds," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 3802–3809.

[67] ——, "Kernel methods on the Riemannian manifold of symmetric positive definite matrices," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 73–80.

[68] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[69] Q.-G. Ji, Z.-D. Fang, Z.-H. Xie, and Z.-M. Lu, "Video abstraction based on the visual attention model and online clustering," *Signal Processing: Image Communication*, vol. 28, no. 3, pp. 241–253, 2013.

[70] Z. Jiang, Z. Lin, and L. Davis, "Recognizing human actions by learning and matching shape-motion prototype trees," *Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 533–547, 2012.

[71] T. Joachims, "Optimizing search engines using clickthrough data," in *International Conference on Knowledge Discovery and Data Mining*, 2002, pp. 133–142.

[72] ——, "Training linear SVMs in linear time," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006, pp. 217–226.

[73] I. Jolliffe, *Principal Component Analysis*. John Wiley & Sons, Ltd, 2002.

[74] S. E. Kahou, V. Michalski, K. Konda, and R. M. C. Pal, "Recurrent neural networks for emotion recognition in video," in *International Conference on Multimodal Interaction (ICMI)*, 2015, pp. 467–474.

[75] A. Karpathy, J. Johnson, and L. Fei-Fei, "Visualizing and understanding recurrent networks," in *International Conference on Learning Representations (ICLR) - Workshop*, 2016.

[76] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[77] H. Kataoka, K. Hashimoto, K. Iwata, Y. Satoh, N. Navab, S. Ilic, and Y. Aoki, "Extended co-occurrence hog with dense trajectories for fine-grained activity recognition," in *Asian Conference on Computer Vision (ACCV)*, 2014, pp. 336–349.

[78] S.-R. Ke, H. L. U. Thuc, Y.-J. Lee, J.-N. Hwang, J.-H. Yoo, and K.-H. Choi, "A review on video-based human activity recognition," *Computers*, vol. 2, no. 2, p. 88, 2013.

[79] Y. Ke, R. Sukthankar, and M. Hebert, "Event detection in crowded videos," in *International Conference on Computer Vision (ICCV)*, 2007.

[80] M. H. Kiapour, K. Yamaguchi, A. C. Berg, and T. L. Berg, "Hipster wars: Discovering elements of fashion styles," in *European Conference on Computer Vision (ECCV)*, 2014, pp. 472–488.

[81] M. Kolar, M. Hradis, and P. Zemcik, "Deep learning on small datasets using online image search," in *Spring Conference on Computer Graphics*, 2016.

[82] H. S. Koppula and A. Saxena, "Learning spatio-temporal structure from RGBD videos for human activity detection and anticipation," in *International Conference on Machine Learning*, 2013.

[83] A. K. S. Kushwaha, O. Prakash, A. Khare, and M. H. Kolekar, "Rule based human activity recognition for surveillance system," in *International Conference on Intelligent Human Computer Interaction (IHCI)*, 2012, pp. 1–6.

[84] Z.-Z. Lan, Y. Yang, N. Ballas, S.-I. Yu, and A. Haputmann, "Resource constrained multimedia event detection," in *International Conference on MultiMedia Modeling*, 2014, pp. 388–399.

[85] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.

[86] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[87] J. Lasserre and C. M. Bishop, "Generative or discriminative? Getting the best of both worlds," in *Bayesian Statistics*, J. Bernardo, M. Bayarri, J. Berger, A. Dawid, D. Heckerman, A. Smith, and M. West, Eds., 2007, vol. 8, pp. 3–24.

[88] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[89] C. P. Lee and C. J. Lin, "Large-scale linear RankSVM," *Neural Computation*, vol. 26, no. 4, pp. 781–817, 2014.

[90] Q. Li, Z. Qiu, T. Yao, T. Mei, Y. Rui, and J. Luo, "Action recognition by learning deep multi-granular spatio-temporal video representation," in *ICMR*, 2016.

[91] Z. Li, Y. Liu, R. Hayward, and R. Walker, "Color and texture feature fusion using kernel PCA with application to object-based vegetation species classification," in *IEEE International Conference on Image Processing (ICIP)*, 2010, pp. 2701–2704.

[92] W. Lin, M.-T. Sun, R. Poovandran, and Z. Zhang, "Human activity recognition for video surveillance," in *International Symposium on Circuits and Systems (ISCAS)*, 2008, pp. 2737–2740.

[93] Z. Lin and J. Brandt, "A local bag-of-features model for large-scale object retrieval," in *European Conference on Computer Vision (ECCV)*, ser. Lecture Notes in Computer Science, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Springer Berlin Heidelberg, 2010, vol. 6316, pp. 294–308.

[94] L. Liu, L. Wang, and X. Liu, "In defense of soft-assignment coding," in *International Conference on Computer Vision (ICCV)*, 2011, pp. 2486–2493.

[95] M. Liu, R. Wang, S. Shan, and X. Chen, "Learning mid-level words on Riemannian manifold for action recognition," *arXiv:1511.04808*, 2015.

[96] D. Lowe, "Object recognition from local scale-invariant features," in *International Conference on Computer Vision (ICCV)*, vol. 2, 1999, pp. 1150–1157 vol.2.

[97] Y. M. Lui, "Human gesture recognition on product manifolds," *Journal of Machine Learning Research*, vol. 13, no. 1, pp. 3297–3321, 2012.

[98] ——, "Tangent bundles on special manifolds for action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 6, pp. 930–942, 2012.

[99] Y. M. Lui and J. Beveridge, "Tangent bundle for human action recognition," in *International Conference on Automatic Face Gesture Recognition and Workshops*, 2011, pp. 97–102.

[100] F. Lv and R. Nevatia, "Recognition and segmentation of 3-d human action using hmm and multi-class adaboost," in *European Conference on Computer Vision (ECCV)*, 2006, pp. 359–372.

[101] M. A. Mendoza and N. Pérez de la Blanca, "HMM-based action recognition using contour histograms," in *Pattern Recognition and Image Analysis*, ser. Lecture Notes in Computer Science, 2007, vol. 4477, pp. 394–401.

[102] A. G. Money and H. Agius, "Video summarisation: A conceptual framework and survey of the state of the art," *Journal of Visual Communication and Image Representation*, vol. 19, no. 2, pp. 121 – 143, 2008.

[103] P. Mundur, Y. Rao, and Y. Yesha, "Keyframe-based video summarization using Delaunay clustering," *International Journal on Digital Libraries*, vol. 6, no. 2, pp. 219–232, 2006.

[104] O. V. R. Murthy and R. Goecke, "Ordered trajectories for large scale human action recognition," in *International Conference on Computer Vision Workshops (ICCVW)*, 2013, pp. 412–419.

[105] M. N. Murty and V. S. Devi, "Nearest neighbour based classifiers," in *Pattern Recognition*, ser. Undergraduate Topics in Computer Science. Springer London, 2011, pp. 48–85.

[106] J. Y.-H. Ng, M. Hausknech, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[107] H. A. Nguyen and J. Meunier, "Gait analysis from video: Camcorders vs. kinect," in *International Conference on Image Analysis and Recognition (ICIAR)*, 2014, pp. 66–73.

[108] B. Ni, X. Yang, and S. Gao, "Progressively parsing interactional objects for fine grained action detection," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[109] W. Niu, J. Long, D. Han, and Y.-F. Wang, "Human activity detection and recognition for video surveillance," in *International Conference on Multimedia and Expo (ICME)*, vol. 1, 2004, pp. 719–722 Vol.1.

[110] J. Oh, Q. Wen, J. Lee, and S. Hwang, "Video abstraction," in *Video Data Management and Information Retrieval*. Idea Group Inc. and IRM Press, 2004, pp. 321–346.

[111] S. O'Hara and B. Draper, "Scalable action recognition with a subspace forest," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012, pp. 1210–1217.

[112] D. Oneata, J. Verbeek, and C. Schmid, "Action and event recognition with Fisher vectors on a compact feature set," in *International Conference on Computer Vision (ICCV)*, 2013, pp. 1817–1824.

[113] J.-Q. Ouyang and R. Liu, "Ontology reasoning scheme for constructing meaningful sports video summarisation," *IET Image Processing*, vol. 7, no. 4, pp. 324–334, 2013.

[114] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *European Symposium on Security and Privacy (EuroS P)*, 2016, pp. 372–387.

[115] O. M. Parkhi, K. Simonyan, A. Vedaldi, and A. Zisserman, "A compact and discriminative face track descriptor," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[116] X. Peng, L. Wang, X. Wang, and Y. Qiao, "Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice," *Computer Vision and Image Understanding (CVIU)*, 2016.

[117] X. Peng, C. Zou, Y. Qiao, and Q. Peng, "Action recognition with stacked Fisher vectors," in *European Conference on Computer Vision (ECCV)*, 2014, vol. 8693, pp. 581–595.

[118] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard*. New York: Van Nostrand Reinhold, 1993.

[119] Ó. Pérez, M. Piccardi, J. García, and J. M. Molina, "Comparison of classifiers for human activity recognition," in *Nature Inspired Problem-Solving Methods in Knowledge Engineering*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, vol. 4528, pp. 192–201. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-73055-2_21

[120] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher kernel for large-scale image classification," in *European Conference on Computer Vision (ECCV)*, 2010, pp. 143–156.

[121] H. Pirsiavash, C. Vondrick, and A. Torralba, *European Conference on Computer Vision (ECCV)*. Springer International Publishing, 2014, ch. Assessing the Quality of Actions, pp. 556–571.

[122] L. Pishchulin, M. Andriluka, and B. Schiele, "Fine-grained activity recognition with holistic and pose based features," in *German Conference on Pattern Recognition (GCPR)*, 2014, pp. 678–689.

[123] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*, 1999, pp. 61–74.

[124] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976 – 990, 2010.

[125] T. Qin, T.-Y. Liu, J. Xu, and H. Li, "Letor: A benchmark collection for research on learning to rank for information retrieval," *Information Retrieval*, vol. 13, no. 4, pp. 346–374, 2010.

[126] V. Ramanathan, C. Li, J. Deng, W. Han, Z. Li, K. Gu, Y. Song, S. Bengio, C. Rossenberg, and L. Fei-Fei, "Learning semantic relationships for better action retrieval in images," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1100–1109.

[127] M. Ramezani and F. Yaghmaee, "A review on human action analysis in videos for retrieval applications," *Artificial Intelligence Review*, pp. 1–30, 2016.

[128] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," in *Conference on Computer Vision and Pattern Recognition (CVPR) - DeepVision Workshop*, 2014.

[129] K. K. Reddy and M. Shah, "Recognizing 50 human action categories of web videos," *Machine Vision and Applications Journal (MVAP)*, pp. 971–981, 2013.

[130] V. Reddy, C. Sanderson, and B. C. Lovell, "Improved foreground detection via block-based classifier cascade with probabilistic decision integration," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 1, pp. 83–93, 2013.

[131] D. Reynolds, "Gaussian Mixture Models," in *Encyclopedia of Biometrics*, S. Li and A. Jain, Eds., 2009, pp. 659–663.

[132] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1, pp. 19 – 41, 2000.

[133] M. Rodriguez, J. Ahmed, and M. Shah, "Action MACH a spatio-temporal maximum average correlation height filter for action recognition," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.

[134] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele, "A database for fine grained activity detection of cooking activities," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 1194–1201.

[135] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the Fisher vector: Theory and practice," *International Journal of Computer Vision*, vol. 105, no. 3, pp. 222–245, 2013.

[136] C. Sanderson and R. Curtin, "Armadillo: a template-based C++ library for linear algebra," *Journal of Open Source Software*, vol. 1, p. 26, 2016.

[137] C. Sanderson and B. C. Lovell, "Multi-region probabilistic histograms for robust and scalable identity inference," in *Lecture Notes in Computer Science (LNCS), Vol. 5558*, 2009, pp. 199–208.

[138] A. Sanin, C. Sanderson, M. Harandi, and B. C. Lovell, "Spatio-temporal covariance descriptors for action and gesture recognition," in *Workshop on the Applications of Computer Vision (WACV)*, 2013, pp. 103–110.

[139] A. Sanin, C. Sanderson, M. Harandi, and B. Lovell, "Spatio-temporal covariance descriptors for action and gesture recognition," in *Workshop on Applications of Computer Vision (WACV)*, 2013, pp. 103–110.

[140] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *International Conference on Pattern Recognition (ICPR)*, vol. 3, 2004, pp. 32–36.

[141] L. Shao, L. Ji, Y. Liu, and J. Zhang, "Human action segmentation and recognition via motion and shape analysis," *Pattern Recognition Letters*, pp. 438 – 445, 2012.

[142] Q. Shi, L. Wang, L. Cheng, and A. Smola, "Discriminative human action segmentation and recognition using semi-Markov model," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.

[143] M. Shimosaka, T. Mori, and T. Sato, "Robust action recognition and segmentation with multi-task conditional random fields," in *International Conference on Robotics and Automation*, 2007, pp. 3780–3786.

[144] S. Shirazi, M. Harandi, B. Lovell, and C. Sanderson, "Object tracking via non-Euclidean geometry: A Grassmann approach," in *Winter Conference on the Applications of Computer Vision (WACV)*, 2014.

[145] S. Shirazi, M. Harandi, C. Sanderson, A. Alavi, and B. Lovell, "Clustering on Grassmann manifolds via kernel embedding with application to action analysis," in *International Conference on Image Processing (ICIP)*, 2012, pp. 781–784.

[146] S. Shirazi, A. Alavi, M. Harandi, and B. Lovell, "Graph-embedding discriminant analysis on Riemannian manifolds for visual recognition," in *Graph Embedding for Pattern Analysis*. Springer New York, 2013, pp. 157–175.

[147] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep Fisher networks for large-scale image classification," in *Advances in Neural Information Processing Systems 26*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds., 2013, pp. 163–171.

[148] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems 27*, 2014, pp. 568–576.

[149] K. Soomro and A. R. Zamir, *Computer Vision in Sports*, 2014, ch. Action Recognition in Realistic Sports Videos, pp. 181–208.

[150] E. H. Spriggs, F. D. L. Torre, and M. Hebert, "Temporal segmentation and activity classification from first-person sensing," in *Conference on Computer Vision and Pattern Recognition (CVPR) - Workshop on Egocentric Vision*, 2009.

[151] D. Sussillo and O. Barak, "Opening the black box: Low-dimensional dynamics in high-dimensional recurrent neural networks," *Neural Computation*, vol. 25, no. 3, pp. 626–649, 2013.

[152] J. Tang, L. Shao, and X. Zhen, "Human action retrieval via efficient feature matching," in *International Conference on Advanced Video and Signal Based Surveillance*, 2013, pp. 306–311.

[153] L. Tao, A. Paiement, D. Damen, M. Mirmehdi, and S. Hannuna, "A comparative study of pose representation and dynamics modelling for online motion quality assessment," *Computer Vision and Image Understanding*, vol. 148, pp. 136–152, 2016.

[154] S. Theodoridis and K. Koutroumbas, "Chapter 2 - classifiers based on Bayes decision theory," in *Pattern Recognition*, 4th ed., S. Theodoridis and K. Koutroumbas, Eds. Boston: Academic Press, 2009, pp. 13 – 89.

[155] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *International Conference on Computer Vision (ICCV)*, 2015.

[156] I. Traore, I. Traore, and A. A. E. Ahmed, *Continuous Authentication Using Biometrics: Data, Models, and Metrics*, 1st ed. Hershey, PA, USA: IGI Global, 2011.

[157] B. T. Truong and S. Venkatesh, "Video abstraction: a systematic review and classification," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 3, no. 1, Feb. 2007.

[158] P. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa, "Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2273–2286, 2011.

[159] P. Turaga and R. Chellappa, "Nearest-neighbor search algorithms on non-Euclidean manifolds for computer vision applications," in *Proceedings of the Seventh Indian Conference on Computer Vision, Graphics and Image Processing*. ACM, 2010, pp. 282–289.

[160] J. C. van Gemert, J.-M. Geusebroek, C. J. Veenman, and A. W. M. Smeulders, *European Conference on Computer Vision (ECCV)*, 2008, ch. Kernel Codebooks for Scene Categorization, pp. 696–709.

[161] R. Vemulapalli, J. Pillai, and R. Chellappa, "Kernel learning for extrinsic classification of manifold features," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 1782–1789.

[162] V. Venkataraman, I. Vlachos, and P. Turaga, "Dynamical regularity for action analysis," in *British Machine Vision Conference (BMVC)*, 2015.

[163] S. N. Vitaladevuni, V. Kellokumpu, and L. Davis, "Action recognition using ballistic dynamics," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.

[164] D. B. Wagner, "Dynamic programming," *The Mathematica Journal*, pp. 42–51, 1995.

[165] B. Wang, Y. Hu, J. Gao, Y. Sun, and B. Yin, "Low rank representation on Grassmann manifolds," in *Computer Vision – ACCV 2014*, ser. Lecture Notes in Computer Science, D. Cremers, I. Reid, H. Saito, and M.-H. Yang, Eds. Springer International Publishing, 2015, vol. 9003, pp. 81–96.

[166] F. Wang, E. Stone, M. Skubic, J. M. Keller, C. Abbott, and M. Rantz, "Towards a passive low-cost in-home gait assessment system for older adults," *IEEE Journal Of Biomedical And Health Informatics*, vol. 17, pp. 346–355, 2013.

[167] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International Journal of Computer Vision*, vol. 103, no. 1, pp. 60–79, 2013.

[168] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *International Conference on Computer Vision (ICCV)*, 2013.

[169] H. Wang, M. M. Ullah, A. Klaser, I. Laptev, and C. Schmid, "Evaluation of local spatio-temporal features for action recognition," in *British Machine Vision Conference (BMVC)*, 2009, pp. 124.1–124.11.

[170] J. Wang, M. She, S. Nahavandi, and A. Kouzani, "A review of vision-based gait recognition methods for human identification," in *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, 2010, pp. 320–327.

[171] L. Wang, Y. Qiao, and X. Tang, "Mining motion atoms and phrases for complex action recognition," in *International Conference on Computer Vision (ICCV)*, 2013, pp. 2680–2687.

[172] ——, "Latent hierarchical model of temporal structure for complex activity classification," *Image Processing*, vol. 23, no. 2, pp. 810–822, 2014.

[173] R. Wang, H. Guo, L. Davis, and Q. Dai, "Covariance discriminative learning: A natural and efficient approach to image set classification," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 2496–2503.

[174] X. Wang, L. Wang, and Y. Qiao, "A comparative study of encoding, pooling and normalization methods for action recognition," in *Asian Conference on Computer Vision (ACCV)*, 2013, pp. 572–585.

[175] D. Weinland, R. Ronfard, and E. Boyer, "A survey of vision-based methods for action representation, segmentation and recognition," *Computer Vision and Image Understanding*, vol. 115, no. 2, pp. 224 – 241, 2011.

[176] D. Wu and L. Shao, "Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 724–731.

[177] J. Wu, Y. Zhang, and W. Lin, "Towards good practices for action video encoding," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 2577–2584.

[178] Y. Wu, Y. Jia, P. Li, J. Zhang, and J. Yuan, "Manifold kernel sparse representation of symmetric positive-definite matrices and its applications," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3729–3741, 2015.

[179] Z. Xu, R. Hu, J. Chen, H. Chen, and H. Li, *Global Contrast Based Salient Region Boundary Sampling for Action Recognition*. Springer International Publishing, 2016, pp. 187–198.

[180] J. Yang, K. Y. andYihong Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 1794–1801.

[181] S.-P. Yong, J. Deng, and M. Purvis, "Key-frame extraction of wildlife video based on semantic context modeling," in *International Joint Conference on Neural Networks (IJCNN)*, 2012, pp. 1–8.

[182] J. Zhang, L. Wang, L. Zhou, and W. Li, "Learning discriminative Stein kernel for SPD matrices and its applications," *IEEE Transactions on Neural Networks and Learning Systems*, (in press).

[183] K. Zhao, A. Alavi, A. Wiliem, and B. C. Lovell, "Efficient clustering on Riemannian manifolds: A kernelised random projection approach," *Pattern Recognition*, vol. 51, pp. 333 – 345, 2016.

[184] Y. Zhuang, Y. Rui, T. Huang, and S. Mehrotra, "Adaptive key frame extraction using unsupervised clustering," in *IEEE International Conference on Image Processing*, vol. 1, 1998, pp. 866–870.

[185] A. Zia, Y. Sharma, V. Bettadapura, E. L. Sarin, M. A. Clements, and I. Essa, *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015, ch. Automated Assessment of Surgical Skills Using Frequency Analysis, pp. 430–438.